

SACLANT ASW
RESEARCH CENTRE
MEMORANDUM

A COMPARISON OF SOME SIGNAL-PROCESSING ALGORITHMS
TO SUPPRESS TOW-VESSEL NOISE IN A TOWED ARRAY, WITH RESULTS
FROM A SHALLOW-WATER TRIAL

by

Ulrich E. RUPE and Even B. LUNDE

1 OCTOBER 1982

NORTH
ATLANTIC
TREATY
ORGANIZATION

LA SPEZIA, ITALY

This document is unclassified. The information it contains is published subject to the conditions of the legend printed on the inside cover. Short quotations from it may be made in other publications if credit is given to the author(s). Except for working copies for research purposes or for use in official NATO publications, reproduction requires the authorization of the Director of SACLANTCEN.

This document is released to a NATO Government at the direction of the SACLANTCEN subject to the following conditions:

1. The recipient NATO Government agrees to use its best endeavours to ensure that the information herein disclosed, whether or not it bears a security classification, is not dealt with in any manner (a) contrary to the intent of the provisions of the Charter of the Centre, or (b) prejudicial to the rights of the owner thereof to obtain patent, copyright, or other like statutory protection therefor.

2. If the technical information was originally released to the Centre by a NATO Government subject to restrictions clearly marked on this document the recipient NATO Government agrees to use its best endeavours to abide by the terms of the restrictions so imposed by the releasing Government.

Published by



-- 1 - AD NUMBER: A125307
-- 2 - FIELDS AND GROUPS: 9/1, 12/2, 12/5, 25/3
-- 3 - ENTRY CLASSIFICATION: UNCLASSIFIED
-- 5 - CORPORATE AUTHOR: SACLANT ASW RESEARCH CENTRE LA SPEZIA (ITALY)
-- 6 - UNCLASSIFIED TITLE: A COMPARISON OF SOME SIGNAL-PROCESSING
ALGORITHMS TO SUPPRESS TOW-VESSEL NOISE IN A TOWED ARRAY, WITH
RESULTS FROM A SHALLOW-WATER TRIAL.
-- 8 - TITLE CLASSIFICATION: UNCLASSIFIED
-- 9 - DESCRIPTIVE NOTE: MEMORANDUM REPT.,
-- 10 - PERSONAL AUTHORS: RUPE, ULRICH E. ; LUNDE, EVEN B. ;
-- 11 - REPORT DATE: OCT 01, 1982
-- 12 - PAGINATION: 92P HC COST: \$ 5.00
-- 14 - REPORT NUMBER: SACLANTCEN-SM-158
-- 20 - REPORT CLASSIFICATION: UNCLASSIFIED
-- 23 - DESCRIPTORS: *MATHEMATICAL ANALYSIS, *ALGORITHMS, *TOWED
ARRAYS, *SIGNAL PROCESSING, NOISE REDUCTION, BEAM FORMING,
ENVIRONMENTAL TESTS, SHALLOW WATER, MATRICES(MATHEMATICS),
EIGENVALUES, VECTOR ANALYSIS, COMPUTER PROGRAMS, ANTISUBMARINE
WARFARE, NATO
-- 24 - DESCRIPTOR CLASSIFICATION: UNCLASSIFIED
-- 27 - ABSTRACT: A SHORT TRIAL HAS BEEN CONDUCTED IN SHALLOW WATER USING
A TOWED ARRAY AND A NUMBER OF SIGNAL-PROCESSING TECHNIQUES. RESULTS
--
-- ARE PRESENTED ON THE OUTPUTS FROM CONVENTIONAL BEAMFORMING AND
-- MAXIMUM-LIKELIHOOD BEAMFORMING AND ON SOME BASIC TECHNIQUES FOR
-- SUPPRESSING TOWSHIP NOISE. THE SIGNAL PROCESSING PACKAGE IS
-- IMPLEMENTED IN SOFTWARE AND PROVIDES THE VARIOUS FORMS OF
-- PROCESSING BY USING THE ARRAY COVARIANCE MATRIX AND EIGENVALUE/
-- VECTOR ANALYSIS. POSSIBLE FUTURE WORK ON THESE TOPICS IS PROPOSED.
-- 28 - ABSTRACT CLASSIFICATION: UNCLASSIFIED
-- 29 - INITIAL INVENTORY: 12
-- 33 - LIMITATION CODES: 1
-- 35 - SOURCE CODE: 342950
-- 36 - DOCUMENT LOCATION: NTIS
-- 40 - GEOPOLITICAL CODE: IT
-- 41 - TYPE CODE: 9
-- 43 - IAC DOCUMENT TYPE:

SACLANTCEN MEMORANDUM SM-158

NORTH ATLANTIC TREATY ORGANIZATION

SACLANT ASW Research Centre
Viale San Bartolomeo 400, I-19026 San Bartolomeo (SP), Italy.

tel: national 0187 560940
international + 39 187 560940

telex: 271148 SACENT I

A COMPARISON OF SOME SIGNAL-PROCESSING ALGORITHMS
TO SUPPRESS TOW-VESSEL NOISE IN A TOWED ARRAY, WITH RESULTS
FROM A SHALLOW-WATER TRIAL

by

Ulrich E. Rupe and Even B. Lunde

1 October 1982

This memorandum has been prepared within the SACLANTCEN
Systems Research Division as part of Project 02.



L.F. WHICKER
Division Chief

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT	1
INTRODUCTION	1
1 METHOD	4
1.1 Sea Trials	4
1.2 Towed-Array System	4
1.3 The On-Board Processing and Analysis System	6
2 ON-LINE DATA MONITORING	6
2.1 Signal Conditioning Unit	6
2.2 Digital Lowpass Filter Bank	6
2.3 On-Line Processing and Monitoring	6
3 OFF-LINE PROCESSING AND ANALYSIS SYSTEM	11
3.1 General	11
3.2 Conventional Beamforming (CB)	14
3.3 "Maximum-Likelihood" Beamforming (MLB)	14
3.4 Conventional Beamforming plus Modified Eigenvalue Noise Suppression (CB + ME)	14
3.5 "Maximum Likelihood" plus Modified Eigenvalue (ML + ME)	14
3.6 Conventional Beamforming plus Reference Beam Noise Suppression (CB + RB)	15
3.7 "Maximum likelihood" plus Reference Beam Noise Suppression (ML + RB)	15
3.8 Conventional Beamforming plus Reference Hydrophone Noise Suppression (CB + RH)	15
3.9 "Maximum Likelihood" plus Reference Hydrophone Noise Suppression (ML + RH)	16
4 RESULTS	16
4.1 Signal Path from Towship to Array in Shallow-Water	16
4.2 Towship-Noise Suppression	23
4.3 Target Indication and Bearing Estimation	32
CONCLUSIONS	33
ACKNOWLEDGMENTS	36
REFERENCES	36
APPENDIX A - COVARIANCE MATRIX ESTIMATE	37
APPENDIX B - BEAMFORMING AND NOISE SUPPRESSION	45
APPENDIX C - THE RANK OF A SOURCE'S CONTRIBUTION TO THE COVARIANCE MATRIX	51
APPENDIX D - RAY THEORY AND NORMAL-MODE THEORY	53
APPENDIX E - LISTINGS OF THE COMPUTER PROGRAMS USED FOR THE SIGNAL PROCESSING	61

List of FiguresPage

1.	Sea trials.	2-3
2.	Deployment of towed array during trials.	5
3.	Sound-speed profiles during trials.	5
4.	On-board processing and analysis system.	7
5.	Example of line-printer output for conventional beamforming.	10
6.	Tabular output corresponding to line-printer output.	12
7.	Example of mode interference at 334 Hz received with a towed array.	18
8a.	Maximum likelihood beamforming output showing corresponding water depths and arrival angles at the bottom (Run 1).	20
8b.	Maximum likelihood beamforming output showing corresponding water depths and arrival angles at the bottom (Run 3).	21
8c.	Maximum likelihood beamforming output showing corresponding water depths and arrival angles at the bottom (Run 4).	22
9a.	Comparison of noise suppression techniques (Run 1).	25
9b.	Comparison of noise suppression techniques (Run 1).	26
9c.	Comparison of noise suppression techniques (Run 2).	27
9d.	Comparison of noise suppression techniques (Run 2).	28
10a.	Comparison of target-bearing indication by conventional beamforming and maximum likelihood beamforming (low frequency).	34
10b.	Comparison of target-bearing indication by conventional beamforming and maximum likelihood beamforming (high frequency).	35

A COMPARISON OF SOME SIGNAL-PROCESSING ALGORITHMS
TO SUPPRESS TOW-VESSEL NOISE IN A TOWED ARRAY, WITH RESULTS FROM A
SHALLOW-WATER SEA TRIAL

by

Ulrich E. Rupe and Even B. Lunde

ABSTRACT

A short trial has been conducted in shallow water using a towed array and a number of signal-processing techniques. Results are presented on the outputs from conventional beamforming and maximum-likelihood beamforming and on some basic techniques for suppressing towship noise. The signal-processing package is implemented in software and provides the various forms of processing by using the array covariance matrix and eigenvalue/vector analysis. Possible future work on these topics is proposed.

INTRODUCTION

The self-noise of the platform can enter a hull-mounted sonar array from different directions and in various frequency bands, thereby presenting a number of problems in the reception of signals during the initial processes of target detection, classification, and tracking when signal levels are low. The advent of the towed array has minimized these problems considerably and, in essence, reduced them to simply one of a noise source ahead of the array.

However, depending on the characteristics of the towed-array system (the scope of the tow cable, the beamwidth of the array, etc.) and the environmental conditions in the area of operation (depth of water, type of sea bottom, etc.), the towship noise can appear in a number of beams in the ahead $\pm 90^\circ$ sector and cause serious operational limitations.

This memorandum compares a number of signal-processing techniques by describing results obtained from a sea-trial in shallow water using a towed-array system.

The general headings are:

- a. The presence of a number of signal paths from the towship to the array.
- b. The effect of the basic techniques for suppressing towship noise.
- c. Target indication and bearing estimation.

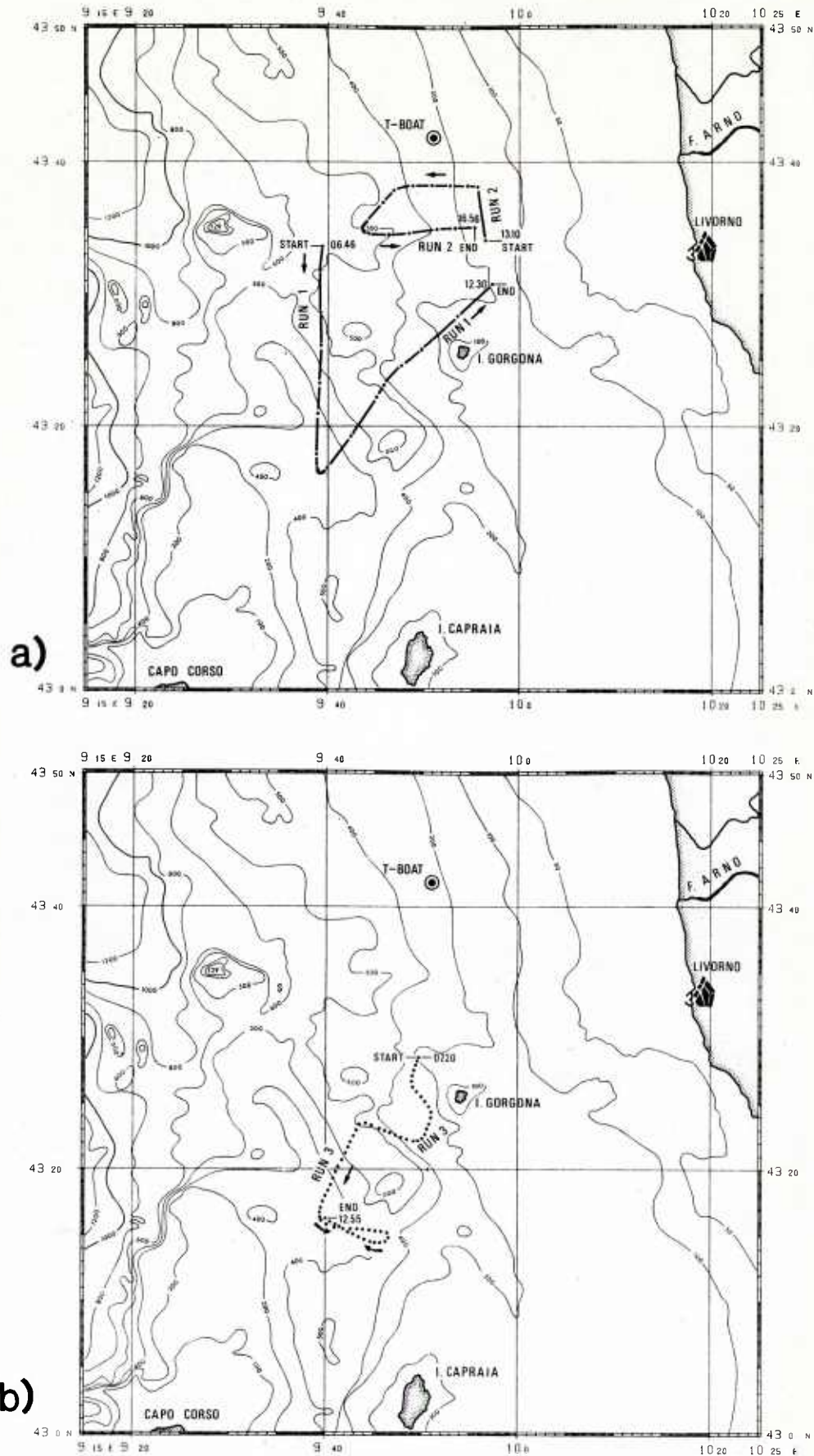


FIG. 1 SEA TRIALS
 a) Runs 1 and 2
 b) Run 3

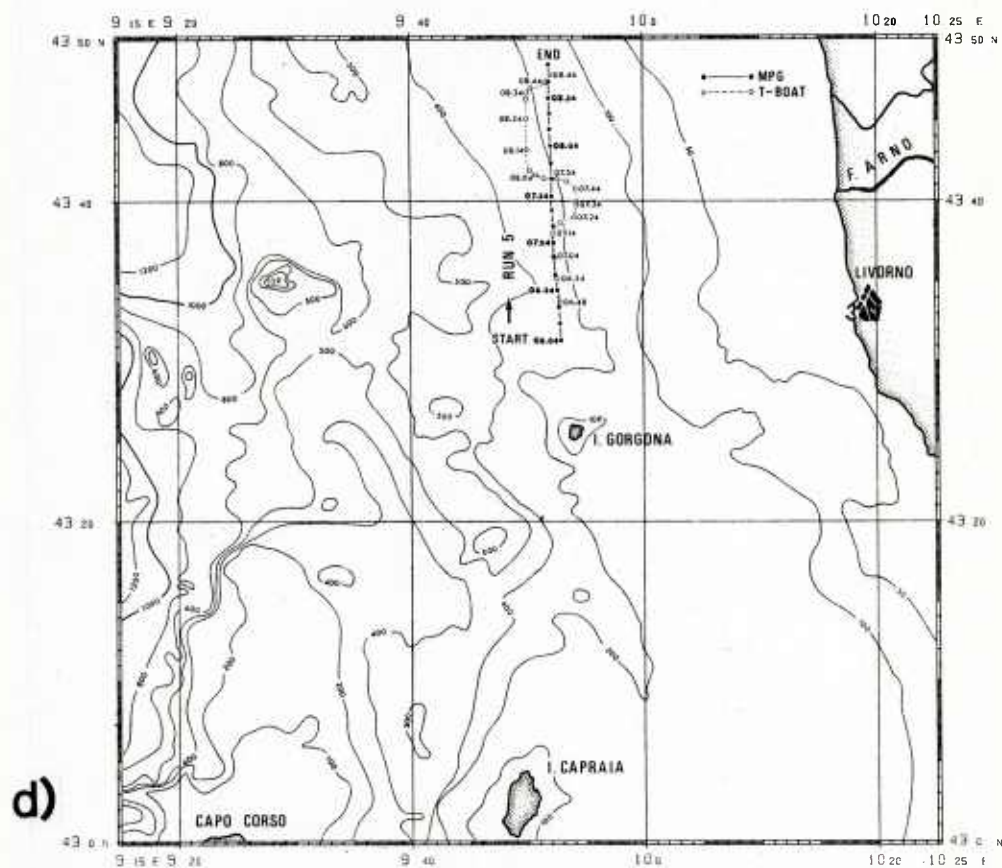
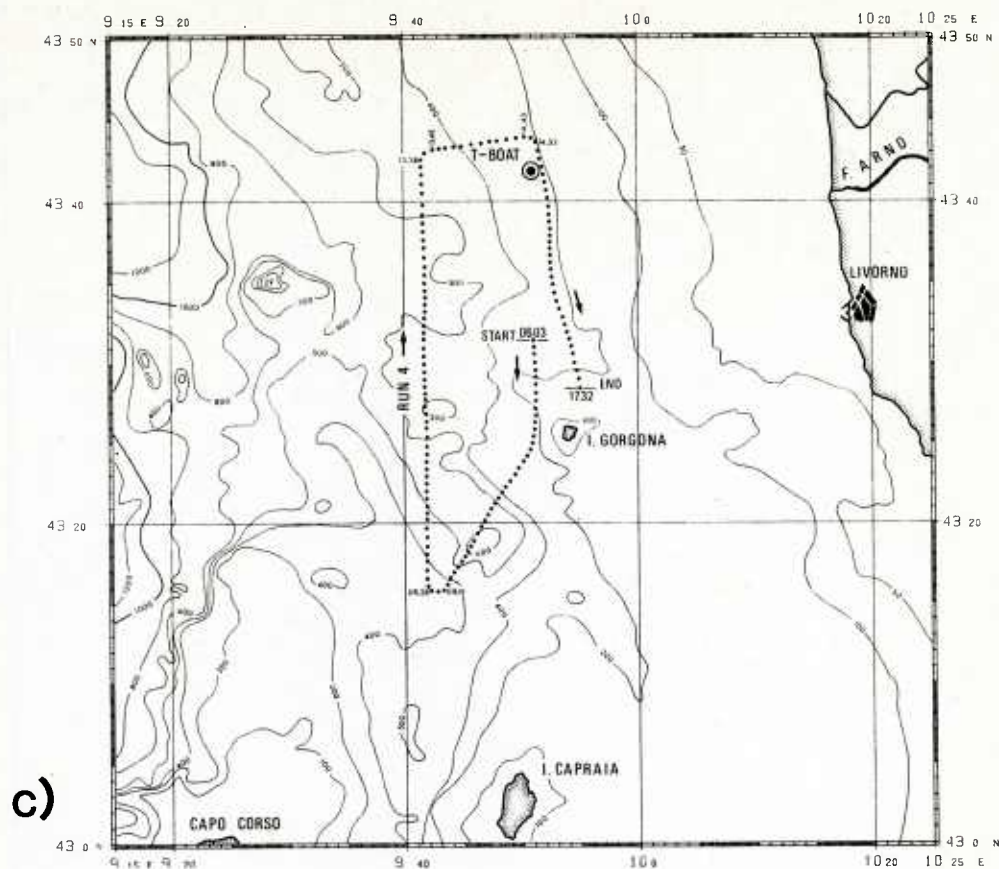


FIG. 1 SEA TRIALS

c) Run 4

d) Run 5

The conclusions and recommendations outline the proposals and arrangements for further studies and sea-trials with the aim of reducing the operational limitations caused by towship noise and improving the estimation of target bearing.

1 METHOD

1.1 Sea Trials

Sea-trials were conducted in the area off the northwestern Italian coast indicated on Fig. 1. Some parts of the area have a constant water depth of 150 m; in other parts the depth increases from 150 m to as much as 500 m. Bottom data are assumed to be the same as those measured by SACLANTCEN in an area a few kilometres further south (Area D1 of <1>). A 3½ m core (Core 127) in that area was mainly of silt with some shells, overlain in the upper 50 cm by sand and then by clay. The relative sound speed was 1.025 and the relative density was about 1.8. No data on bottom-reflection losses are available. For the theoretical predictions of transmission loss in the experimental area, sound-speed data have been taken from <2>.

During the trials the sea-state was SS-0 most of the time and never higher than SS-1. The towed array was deployed from SACLANTCEN's research vessel MARIA PAOLINA G. (MPG) as shown in Fig. 2 and most of the sea-trials were conducted at a speed of 5 kn. SACLANTCEN's workboat MANNING transmitted 334 Hz signals from a source suspended at depths of either 50 or 100 m. These signals were used primarily as a CW source with different source levels.

Figure 1 shows the relative positions of the MARIA PAOLINA G. and MANNING for the five runs constituting the trials. In Runs 1 to 4 (Figs. 1a, b, c) the MANNING was at anchor whilst transmitting; during Run 5 (Fig. 1d) it was underway. The various runs placed the MANNING on all possible bearings and at ranges from 1 km to 30 km. Figure 3 gives the sound-speed profiles recorded during the sea-trial. They show that a shallow sound channel usually existed at about 100 m.

1.2 Towed-Array System

SACLANTCEN's Prakla-Seismic towed array was used, with a maximum tow-cable scope of 700 m. The hydrophone spacing employed was 1.96 m, corresponding to $\lambda/2$ at approximately 385 Hz.

An additional hydrophone, referred to as the reference hydrophone, was used to investigate one of the basic methods of suppressing towship noise. This could be clamped onto the tow-cable at distances of up to 200 m behind the towship (Fig. 2). No streamlined body was available for this additional hydrophone during this sea trial and heavy flow noise was expected to degrade the correlation with the array signals. The signal from the reference hydrophone was brought in-board for use in the signal-processing techniques (see Ch. 3).

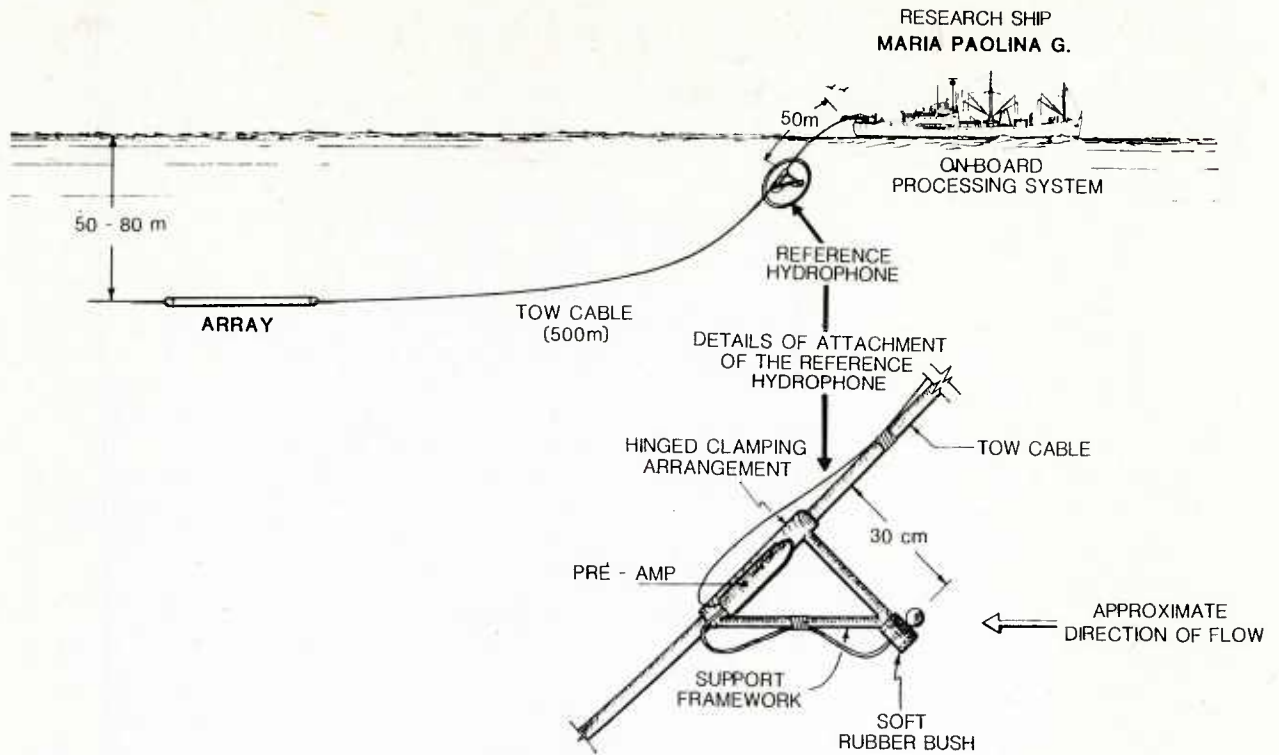


FIG. 2 DEPLOYMENT OF TOWED ARRAY DURING TRIALS

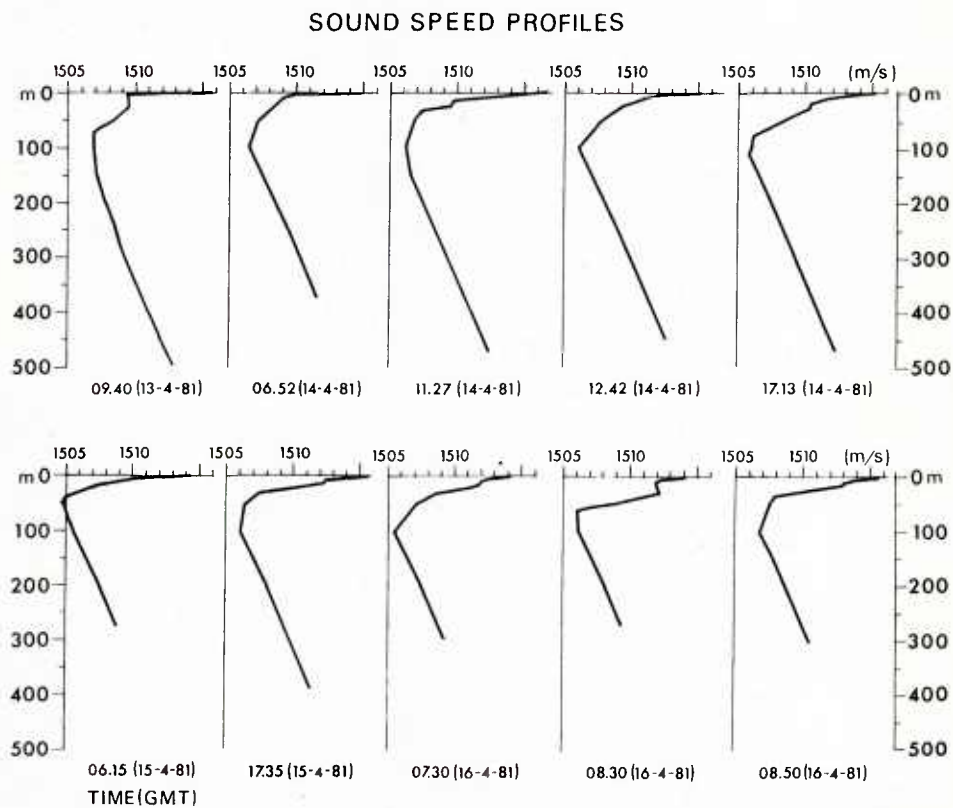


FIG. 3 SOUND-SPEED PROFILES DURING TRIALS

1.3 The On-Board Processing and Analysis System

A block diagram of the on-board processing and analysis system used in the sea-trials is shown in Fig. 4. It consisted of two main parts: the on-line monitoring system and the off-line processing system, as described in more detail in the following chapters.

2 ON-LINE DATA MONITORING

The on-line data monitoring system on board the MPG provided a display of either the output of a conventional beamforming system (CBF) or of the maximum-likelihood method (MLM) of beamforming for a single selected frequency. The display, which used the standard computer line-printer in a novel manner, gave an immediate visual indication of the power in the CBF or MLM beams. The component parts of the system are described below.

2.1 Signal Conditioning Unit

The lowest cutoff frequency of the available anti-aliasing lowpass filter bank was 1 kHz. As the frequency range of interest was below 350 Hz, a sampling frequency of ≥ 2.9 kHz was needed to have an aliasing level of ≤ -40 dB. For reasons given below, the chosen sampling frequency of the A/D converter was 12.288 kHz.

2.2 Digital Lowpass Filter Bank

The Plessey beamformer was used as a lowpass filter bank in order to reduce the sampling frequency by implementing 48-point linear-phase finite-impulse-response (FIR) filters with 3.072 kHz ($= 12.288/4$ kHz) input sampling frequency, and 819.2 Hz ($= 12.288/15$ kHz) output sampling frequency. The beamformer was also used to delay the data from the reference hydrophone.

2.3 On-Line Processing and Monitoring

A MAP array processor and an HP minicomputer were used for on-line processing and monitoring. The first operation was a 1024-point discrete fourier transform (DFT), implemented by a fast fourier transform (FFT), which gave a resolution about 0.8 Hz ($= 819.2/1024$ Hz), and a data window of 1.25 s length. The data collection and FFT processing had to be done in sequence, thereby giving an "acquisition" time of approximately 2.4 s. In other words, a time window of 1.25 s was repeated each 2.4 s, yielding a time gap of about 1.15 s. The DFT outputs for ten selected frequencies were written on magnetic tape for use by the off-line signal-processing system.

Flexibility in the choice of the parameters for the calculation of the covariance matrix was obtained by the scheme shown in Table 1. The covariance matrices are calculated for a selected number of frequencies

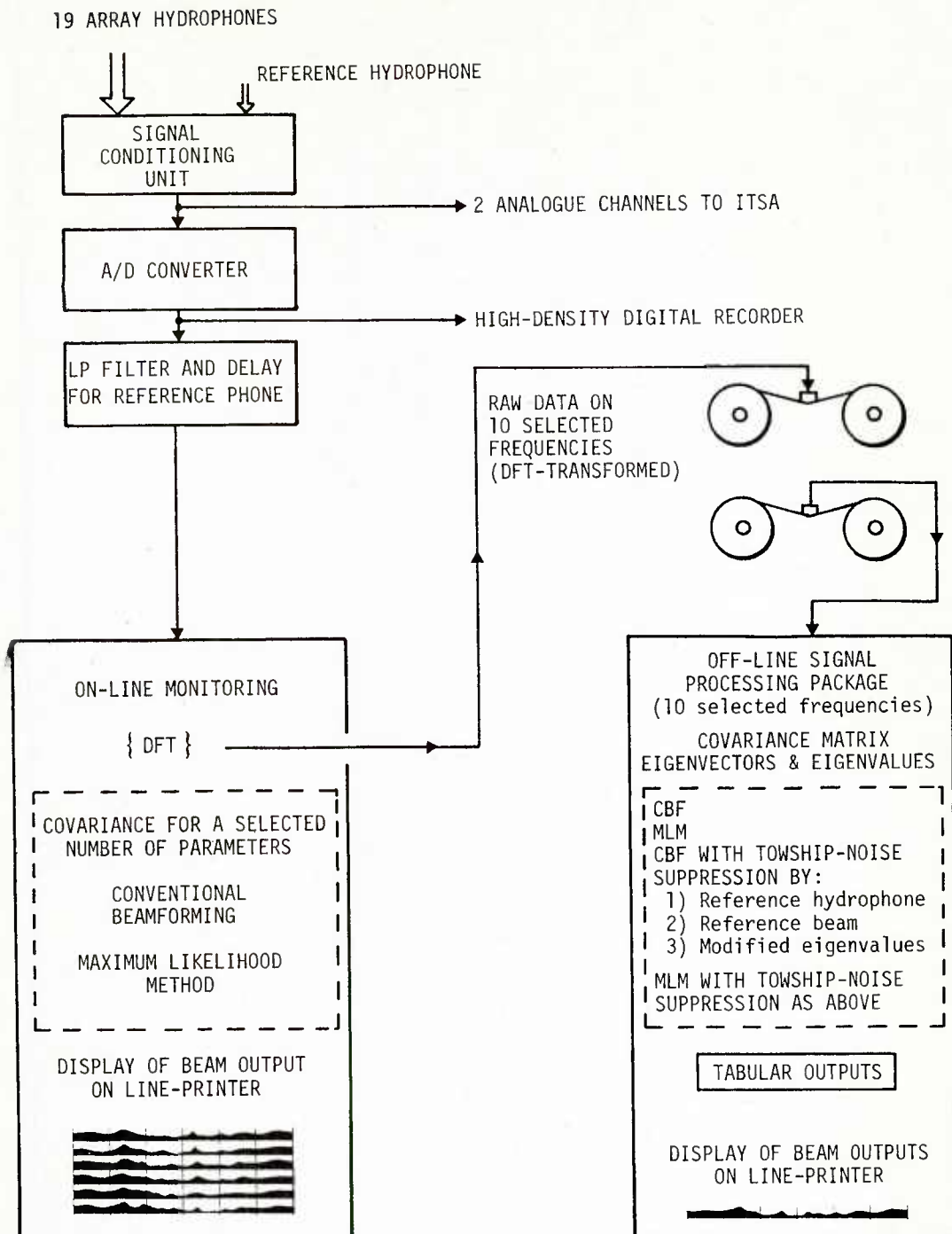


FIG. 4 ON-BOARD PROCESSING AND ANALYSIS SYSTEM

TABLE 1

CALCULATION OF THE NARROW-BAND SPATIAL COVARIANCE MATRIX

DO 50 NS	No. of averages
DO 20 NE	No. of hydrophones
CALL X(T)	Time-domain data for the phone (NE)th
CALL FFT	Frequency-domain data in terms of real and imaginary parts
DO 10 NF	No. of selected frequencies
STORE REAL PART AND IMAGINARY PART FOR EACH FREQUENCY NF AND HYDROPHONE NE	
→ 10 CONTINUE	
→ 20 CONTINUE	
DO 40 NF	No. of selected frequencies
LP = 1	
DO 30 K1 = 1, NE	
DO 30 K2 = 1, NE	
JF(K2, LE, K1) GOTO 30	
AR =	Real part of signal A
AI =	Imaginary part of signal A
BR =	Real part of signal B
BI =	Imaginary part of signal B
RC = AR*BR+AI*BI	
AC = AR*BI-AI*BR	
RCS(NF, LP) = RCS(NF, LP) + RC	
ACS(NF, LP) = ACS(NF, LP) + AC	
LP = LP + 1	
→ 30 CONTINUE	
→ 40 CONTINUE	
→ 50 CONTINUE	
NC = NE *(NE-1)/2	No. of combinations
DO 60 K = 1, NF	
DO 60 L = 1, NC	
RCS(K, L) = RCS(K, L)/NS	
ACS(K, L) = ACS(K, L)/NS	
→ 60 CONTINUE	

from filtered and fourier-transformed time-domain data of each hydrophone channel. The required average time is obtained by repeating this operation.

Thus for one selected frequency out of the ten, the output vector:

$$X_S = \begin{bmatrix} X_{1,S} & X_{2,S} & \dots & X_{N,S} \end{bmatrix}^T$$

was used to build up an estimate of the covariance matrix:

$$R_j = \frac{1}{M} \sum_{i=1}^M X(Mj+i) \cdot X^*(Mj+i) \quad , \quad (\text{Eq. 1})$$

where N is the number of hydrophones used to build up the output vector X_S for the time index S and j is the number of matrices calculated.

The matrix R_j was used to produce the mean power outputs for 181 beams, one for every degree, from one endfire to the other. These outputs were converted to decibel levels and then displayed on a line printer. An example for conventional beamforming is shown in Fig. 5. The bearing is marked above the top-line for a narrow-band processing of 334 Hz. All other outputs were produced with the off-line processing system. Dependent on the number of acquired time-series, such an output was produced for monitoring purpose at intervals of 1 to 2 minutes. The beamforming was either conventional:

$$P_{k,j} = B_k^* R_j B_k \quad k = 1, 2, \dots, 181 \quad (\text{Eq. 2})$$

or "maximum-likelihood":

$$P_{k,j} = \left(B_k^* R_j^{-1} B_k \right)^{-1} \quad k = 1, 2, \dots, 181 \quad , \quad (\text{Eq. 3})$$

where B_k is the steering vector determining the beam direction:

$$B_k = \left[\dots, \exp \left(-j2\pi \frac{x_n}{\lambda} \cos \beta_k \right), \dots \right]^T \quad , \quad (\text{Eq. 4})$$

in which x_n is the coordinate of hydrophone n and β_k is beam direction.

Every time the covariance matrix were calculated, the "acquisition" time was increased by the order of 2 to 3 seconds. Depending on the number of acquisitions, the beam outputs were calculated and printed from the averaged matrix. The minimum number of averages had to equal the number of hydrophones.

CONVENTIONAL BEAM-FORMING

FREQUENCY : 334.40

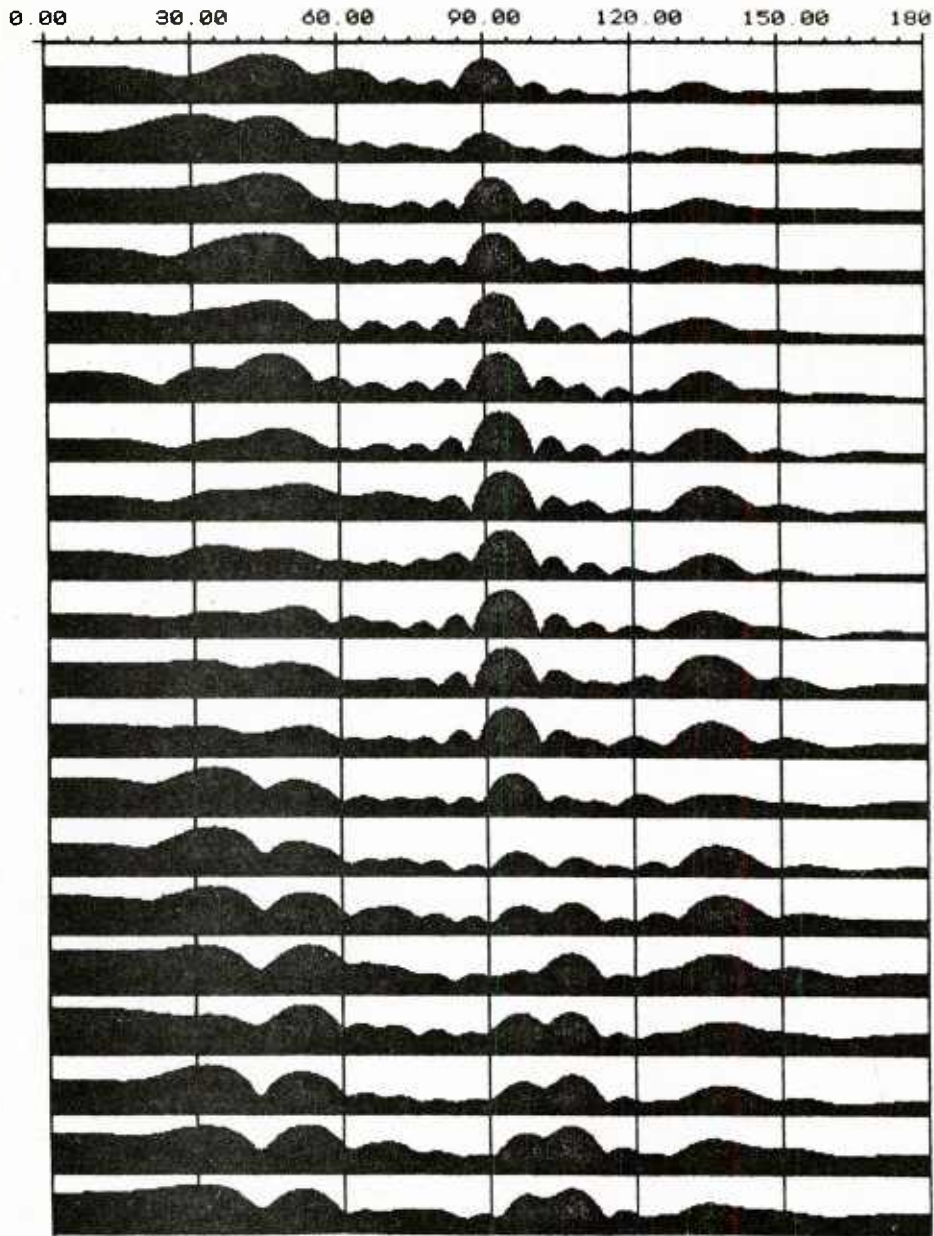


FIG. 5 EXAMPLE OF LINE-PRINTER OUTPUT FOR CONVENTIONAL BEAMFORMING

3 OFF-LINE PROCESSING AND ANALYSIS SYSTEM

The off-line processing and analysis system used tapes of discrete fourier transform (DFT) data produced by the on-line system for ten preselected frequencies and gave, as outputs, the line-printer display mentioned above for the on-line monitoring system and tabulated data relating to the various parameters of interest. The off-line analysis provided the above outputs for the following signal-processing techniques:

- a. Conventional beamforming with no shading (CBF).
- b. Maximum-likelihood method of beamforming (MLB).
- c. CBF and MLB with basic techniques for suppressing towship noise by:
 - 1) Use of a reference hydrophone.
 - 2) Use of a reference beam.
 - 3) Use of modified eigenvalues.

3.1 General

The off-line processing system consisted of an HP minicomputer with vector instruction set (VIS) option <3>, a magnetic tape station, a line printer, and a terminal. The input data from magnetic tape were from 19 array hydrophones and one reference hydrophone, quadrature bandpass-filtered around ten selected frequencies and with a bandwidth of 0.8 Hz. These data were then processed by a signal-processing program, and the output written on disc. A program produced plots of these data on the line printer in the same format as the on-line monitoring program (Fig. 5).

In addition a tabular output was produced, as shown in Fig. 6, for each line-printer output. This tabular output shows a whole range of data; for example, columns 4, 13 and 22 give the beam numbers from 1 to 180, columns 5, 14 and 23 give the beam angles for the frequency in use, and the columns under CON and MLM give the values of $P_{k,j}$ (Eqs. 2 and 3) for the conventional and "maximum-likelihood" beam processing.

The first step in the processing program is to create the following quantities:

$$R_{j,q} = \frac{1}{M} \sum_{i=1}^M x(Mj+i,q) \cdot x^*(Mj+i,q) \quad (\text{Eq. 5})$$

$$Q_{j,q} = \frac{1}{M} \sum_{i=1}^M x(Mj+i,q) \cdot y^*(Mj+i,q) \quad (\text{Eq. 6})$$

$$r_{j,q} = \frac{1}{M} \sum_{i=1}^M y(Mj+i,q) \cdot y^*(Mj+i,q) \quad (\text{Eq. 7})$$

N

5

BEARING

	CON	MLM	SPL	BE1	BE2	BO					
1	89	104	1	-11	1	25	1	-6	1	2	2
2	53	127	2	-11	1	20	1	-10	1	1	1
3	20	127	3	-11	1	13	1	-13	0	1	0
4	10	130	4	-11	1	8	1	-16	0	0	0
5	5	103	5	-11	1	10	1	-16	0	0	2
6	3	90	6	-11	1	22	3	-14	0	1	4
7	2	84	7	-11	1	77	3	-10	1	3	7
8	2	82	8	-11	1	77	3	-6	1	5	11
9	1	96	9	-10	1	115	3	-2	2	7	14
10	1	85	10	-10	1	130	8	-1	2	9	16
11	1	97	11	-10	1	177	15	-1	0	17	0
12	1	118	12	-10	1	182	15	-1	2	9	15
13	1	93	13	12	30	195	56	-19	7	13	0
14	1	104	14	12	31	191	55	-28	1	4	10
15	0	90	15	15	31	188	37	-35	2	2	8
16	0	80	16	18	31	195	26	-36	4	1	7
17	0	91	17	20	31	221	20	-29	8	3	8
18	0	71	18	22	31	270	18	-14	13	8	11
19	0	109	19	24	31	341	19	9	20	15	15
20	127	70	20	26	31	426	23	38	26	23	20
1	43	52	21	27	31	514	33	68	33	35	23
2	24	32	22	29	32	590	58	96	37	44	26
3	15	15	23	30	32	640	130	116	40	50	25
4	114	107	24	32	32	656	222	126	41	53	22
5	148	136	25	33	32	634	108	122	39	52	18
6	33	43	26	34	32	577	46	103	35	48	12
7	101	97	27	36	32	493	25	73	30	40	7
8	169	172	28	37	31	393	16	33	23	31	3
9	140	128	29	38	31	298	12	-12	17	21	0
10	52	60	30	39	31	164	9	-5	11	13	0
11	69	74	31	40	31	110	8	-9	6	6	0
12	74	78	32	42	31	52	7	-123	2	2	1
13	124	115	33	43	31	23	7	-141	0	0	1
14	35	45	34	44	33	24	7	-141	0	1	1
15	58	65	35	45	33	57	7	-125	3	3	1
16	94	92	36	46	33	121	8	-95	7	8	2
17	83	84	37	47	33	213	9	-52	14	13	3
18	91	90	38	48	33	324	11	-2	23	20	6
19	180	-11	39	49	52	444	16	49	33	26	10
20	44	53	40	50	52	558	26	95	42	32	16
1	482	-11	41	51	52	651	55	132	50	37	22
2	16	30	42	52	52	709	189	153	55	40	27
3	3	1	43	52	52	721	230	157	56	40	31
4	18	1	44	53	52	686	55	144	54	38	31
5	1	335	45	54	52	607	21	116	48	33	28
6	0	127	46	55	52	497	11	77	39	27	23
7	3	1	47	56	52	371	7	34	29	20	17
8	6	70	48	57	52	249	9	-8	19	13	10
9	0	20	49	58	52	145	4	-43	11	7	5
10	0	32	50	59	52	70	3	-67	5	3	1
11	1	2	51	60	52	27	3	-78	1	1	0
12	0	0	52	60	52	12	3	-77	0	0	1
13	0	0	53	61	49	17	3	-67	0	1	3
14	1	0	54	62	49	31	3	-51	2	2	6
15	0	0	55	63	49	46	3	-33	3	3	8
16	0	0	56	64	49	54	3	-18	3	3	9
17	0	2	57	64	49	53	3	-6	3	3	8
18	03352		58	65	49	44	3	1	3	2	7
19	01500		59	66	49	31	3	4	2	1	5
20	215265		60	67	49	20	3	3	1	0	2

14

BEARING

	CON	MLM	SPL	BE1	BE2	BO
12	3	1	0	0	2	
10	3	-1	0	0	1	
12	3	1	0	0	1	
17	3	-1	0	1	1	
21	3	0	1	1	2	
23	3	2	1	1	2	
21	3	2	1	1	1	
12	3	-0	0	0	1	
8	3	0	0	0	0	
3	3	-7	0	0	0	
11	4	-8	0	0	0	
14	4	-7	0	1	0	
16	4	-5	1	1	0	
8	2	0	0	0	0	
6	2	-0	0	0	0	
1	2	-0	0	0	0	
1	2	-1	1	0	0	
13	2	0	0	0	0	
9	2	-0	0	0	0	
8	2	-4	0	0	1	
8	2	-6	0	0	2	
9	2	-7	0	0	3	
10	2	-7	0	1	4	
11	2	-5	0	1	4	
13	2	-3	0	0	4	
14	3	-1	0	0	3	
16	4	1	0	0	3	
19	5	2	0	0	2	
6	2	0	0	0	0	
7	0	0	0	0	0	
7	0	-2	0	0	0	
6	0	-6	0	0	0	
8	1	-8	1	0	0	
10	4	-10	1	0	0	
14	0	-8	0	0	4	
18	0	-4	0	0	5	
13	0	0	2	10	17	
22	1	3	12	173	-10	
21	1	3	12	174	-11	
20	1	3	11	175	-11	
16	2	2	10	176	-11	
12	0	2	8	177	-11	
3	0	1	7	178	-11	
0	0	0	6	179	-11	
4	0	0	4	180	-11	

23

BEARING

	CON	MLM	SPL	BE1	BE2	BO
21	5	-13	0	0	3	
19	5	-14	0	0	3	
15	5	-14	0	0	2	
17	5	-12	0	0	1	
20	5	-9	0	0	0	
22	5	-5	0	0	1	
23	5	-2	0	0	1	
22	5	-0	0	0	0	
21	4	0	0	0	0	
17	4	-3	0	0	0	
14	3	-0	0	0	0	
16	3	-5	0	0	0	
16	3	-5	0	0	0	
16	3	-5	0	0	0	
15	3	-4	0	0	0	
15	3	-4	0	0	1	
15	3	-4	0	0	1	
17	6	-5	0	0	1	
23	8	-5	0	0	1	
28	10	-4	0	0	0	
34	13	-2	0	0	0	
41	17	0	0	0	0	
48	22	0	0	0	1	
55	26	5	0	0	3	
26	8	0	0	0	5	
64	24	9	1	1	11	
14	19	1	1	1	14	
56	10	5	1	1	15	
8	2	1	1	15		
39	6	-1	1	0	15	
5	-3	1	0	13		
24	5	-4	0	10		
4	-5	0	1	2		
4	-4	0	1	1		
4	-4	0	1	0		
4	-3	0	1	0		
4	-3	0	1	1		
4	-3	0	1	1		
11	5	-2	0	0	2	
12	5	-2	0	0	2	
15	7	-2	0	0	2	
16	7	-3	0	0	2	
20	8	-3	0	0	1	
21	9	-2	0	0	1	
22	9	-2	0	0	1	
22	10	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0	1	
25	11	-1	0	0</		

where

X : Data vector from array hydrophones
 y : Data scalar from reference hydrophone
 R : Covariance matrix
 Q : "Reference vector"
 r : "Reference scalar"
 q : Frequency index
 i : Acquisition time index
 j : Time index after averaging
 M : Number of acquisitions

The indices are omitted hereafter.

The next step was to obtain the eigenvalues and eigenvectors of R. As shown in App. A, R can be written

$$R = \sum_{n=1}^N \lambda_n E_n E_n^* , \quad (\text{Eq. 8})$$

where λ_n are eigenvalues

E_n are eigenvectors

$n = 1, 2, \dots, N$

and N is the number of the array hydrophones

There were three reasons for introducing the "eigen-quantities". Firstly, these quantities might be of interest on their own. Several signal-processing methods are based on explicit knowledge of these quantities. Only one was implemented in this version of the program, the so-called modified eigenvalue method.

Secondly, by calculating these eigen-quantities, one might gain some feeling for their behaviour under more or less complex sound-field conditions.

Thirdly, as shown in App. A and B, the ten signal-processing modes implemented for this sea trial could all be formulated and calculated in terms of the eigen-quantities. The signal processing in the off-line program was therefore based on the use of the eigen-quantities, whereas the on-line program used a matrix inversion for the maximum-likelihood technique. A check on the two programs could thus be made by comparing the MLM columns of the outputs illustrated in Fig. 6. The following sections describe these methods both with and without the use of eigenvalues and eigenvectors.

3.2 Conventional Beamforming (CB)

The mean power output from an unweighted (unshaded) conventional beam can be written as (Sect. B.1 of App. B):

$$P = B^* R B .$$

Introducing eigenvalues and eigenvectors:

$$P = \sum_{n=1}^N \lambda_n |B^* E_n|^2 . \quad (\text{Eq. 9})$$

3.3 "Maximum-Likelihood" Beamforming (MLB)

The mean power output can in this case be written as (Sect. B.2 of App. B):

$$P = (B^* R^{-1} B)^{-1} , \quad (\text{Eq. 10})$$

and with eigenvalues and eigenvectors:

$$P = \left(\sum_{n=1}^N \lambda_n^{-1} |B^* E_n|^2 \right)^{-1} . \quad (\text{Eq. 11})$$

3.4 Conventional beamforming plus modified eigenvalue noise suppression (CB + ME)

The eigenvalues of R were modified according to some rules. One such rule was to make eigenvalue λ_i equal to the smallest eigenvalue if the corresponding eigenvalue E_i correlated well with the steering vector

representing the direction of the towship. Then:

$$P = \sum_{n=1}^N \lambda_n^{-1} |B^* E_n|^2 . \quad (\text{Eq. 12})$$

3.5 Maximum likelihood plus modified eigenvalue (ML + ME)

This is a combination of ML beamforming with modified eigenvalues

$$P = \left(\sum_{n=1}^N \lambda_n^{-1} |B^* E_n|^2 \right)^{-1} . \quad (\text{Eq. 13})$$

3.6 Conventional beamforming plus reference beam noise suppression (CB+RB)

A reference beam direction with steering vector B_r is chosen according to some rule and this is used to calculate (Sect. B.3 of App. B):

$$P = (B^*RB) \left[1 - |B^*RB_r|^2 (B^*RB \cdot B_r^* R B_r)^{-1} \right]. \quad (\text{Eq. 14})$$

And with eigen-quantities:

$$B^* R B = \sum_{n=1}^N \lambda_n |B^* E_n|^2, \quad (\text{Eq. 15})$$

$$B_r^* R B_r = \sum_{n=1}^N \lambda_n |B_r^* E_n|^2, \quad (\text{Eq. 16})$$

$$B^* R B_r = \sum_{n=1}^N \lambda_n (B^* E_n) (B_r^* E_n)^* . \quad (\text{Eq. 17})$$

3.7 "Maximum likelihood" plus reference beam noise suppression (ML + RB)

Also, from Sect. B.3 of App. B, we have:

$$P = (B^* R^{-1} B)^{-1} \left[1 - |B^* R^{-1} B_r|^2 (B^* R^{-1} B \cdot B_r^* R^{-1} B_r)^{-1} \right]. \quad (\text{Eq. 18})$$

And with eigen-quantities:

$$B^* R^{-1} B = \sum_{n=1}^N \lambda_n^{-1} |B^* E_n|^2, \quad (\text{Eq. 19})$$

$$B_r^* R^{-1} B_r = \sum_{n=1}^N \lambda_n^{-1} |B_r^* E_n|^2, \quad (\text{Eq. 20})$$

$$B^* R^{-1} B_r = \sum_{n=1}^N \lambda_n^{-1} (B^* E_n) (B_r^* E_n)^* . \quad (\text{Eq. 21})$$

3.8 Conventional beamforming plus reference hydrophone noise suppression (CB + RH)

The mean output power in this mode is (Sect. B.4 of App. B):

$$P = B^* R B \cdot \left[1 - |B^* Q|^2 (B^* R B \cdot r)^{-1} \right] \quad (\text{Eq. 22})$$

With eigenvector and eigenvalues, this was implemented using:

$$B^* R B = \sum_{n=1}^N \lambda_n |B^* E_n|^2, \quad (\text{Eq. 23})$$

$$B^* Q = \sum_{n=1}^N (B^* E_n) (Q^* E_n)^* \quad (\text{Eq. 24})$$

3.9 "Maximum Likelihood" plus reference-hydrophone noise suppression (ML + RH)

With reference to Sect. B.4 of App. B:

$$P = (B^* R^{-1} B)^{-1} \left[1 + |B^* R^{-1} Q|^2 \{B^* R^{-1} B (r - Q^* R^{-1} Q)\}^{-1} \right] \quad (\text{Eq. 25})$$

With eigen-quantities

$$B^* R^{-1} B = \sum_{n=1}^N \lambda_n^{-1} |B^* E_n|^2, \quad (\text{Eq. 26})$$

$$B^* R^{-1} Q = \sum_{n=1}^N \lambda_n^{-1} (B^* E_n) (Q^* E_n)^*, \quad (\text{Eq. 27})$$

$$Q^* R^{-1} Q = \sum_{n=1}^N \lambda_n^{-1} |Q^* E_n|^2 \quad (\text{Eq. 28})$$

4 RESULTS

This chapter presents and discusses sample data from all runs except Run 2, which are similar to those from Runs 1, 3 and 4. Some data are presented from a previous sea-trial to show the interference effects in the signal paths from a remotely deployed transducer to an array.

4.1 Signal path from towship to array in shallow-water

The geometry of the trials is shown on the left of Table 2. Typical geometry with an operational towed array is, however, more likely to be as shown on the right of the table.

TABLE 2

GEOMETRY OF TRIALS AND TYPICAL GEOMETRY

	<u>Trials</u>	<u>Typical</u>
Source depth	3 m	2 to 5 m
Tow-cable scope	200 to 700 m	500 to 2000 m
Array length	40 m	50 to 200 m
Array depth	50 to 100 m	-
Water depth	100 to 500 m	-

Together with the type of sea-bottom, the sea-state, and the sound-speed profile, the geometry defines the towship's noise field at an array towed in shallow water. It will thus be appreciated that the noise field at the towed array can be very complex and difficult to predict. Thus care must be taken in using any of the various available techniques.

Ray-tracing and normal-mode propagation are derived theoretically in App. D, indicating that ray-tracing cannot describe accurate low-frequency propagation in shallow water. For ray-tracing there is a general rule of thumb that the water depth should be greater than 10λ . Table 3 indicates some safe limits for the use of ray tracing.

TABLE 3

MINIMUM WATER DEPTH (10λ) FOR SAFE USE OF RAY TRACING

<u>Frequency</u> (Hz)	<u>Minimum depth</u> (m)
50	300
100	150
250	60
500	30

Thus, for example, ray tracing for frequencies of 100 Hz should not be used in water shallower than 150 m; equally, in a water depth of 150 m ray tracing should not be used for frequencies below 100 Hz.

In the case of the normal-mode solution there is a difficulty at short range because of the incomplete description of the acoustical field in terms of the normal-mode solution, which is valid for farfield only. It is usual to discard the short-range continuous field. Thus there is a minimum range below which the discrete modes alone do not sufficiently describe the field. This minimum range is usually taken as 3 to 5 water depths. Thus in 150 m water depth, ray tracing would be satisfactory at frequencies above 100 Hz out to ranges of between 450 to 750 m. Beyond this range the normal-mode solution should be used, particularly for shallow water. As, in fact, different modes travel with different velocities and arrive

at different angles, a mismatch of the array to the acoustical field might become a serious problem for signal processing at low frequencies. This is evidently true for the endfire direction of a horizontal array <5>. A short explanation of the need to distinguish between rays and modes for array processing at low frequencies in shallow water can be given readily by use of Eq. 4, this being the steering vector determining the beam direction. The factor $2\pi x_n/\lambda$ in that equation, which is essentially the array aperture size in wave length, can be rewritten as $\omega x_n/c$, where c is the phase velocity at the receiving hydrophone and is a constant value at one receiver for all the rays impinging on this receiver. However for all modes exhibiting their individual phase velocities, this conventional array processing will be matched to one velocity only, hence with increasing dispersion the array will become increasingly mismatched.

It is not the intention of this present memorandum to make an extensive analysis of the towship's noise field at the array (such an analysis should probably use the fast field program or the range-dependent parabolic equation techniques <4>), but merely to show that the noise field is complex, with a covariance matrix of rank >1. This is the same as saying that the noise from one source arrives from different directions at the same receiver when added coherently. The resolution of various paths from the towship with the array was shown in <5>, using different signal-processing techniques. With those methods and a fairly good signal-to-noise ratio, only four phones of a towed array were used to resolve the towship's noise directions.

An additional indication of the complex nature of acoustic signals in shallow water was obtained as a side result of a previous sea trial. Figure 7 gives an experimental example of mode interference over a signal path of 3 to 4 km. The range from the transmitting site to the receiving

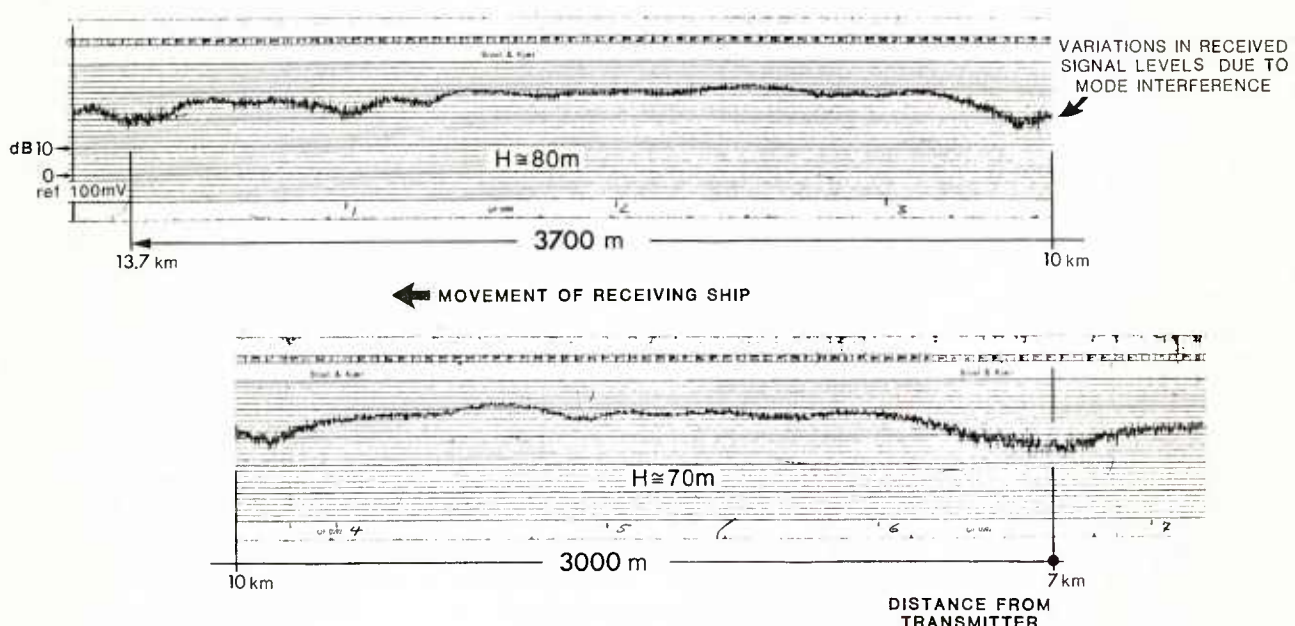


FIG. 7 EXAMPLE OF MODE INTERFERENCE AT 334 Hz
RECEIVED WITH A TOWED ARRAY
(taken from earlier experiments)

ship varied from 7 to about 14 km during the measurements. For this distance the modes can be assumed to be well separated. A strong, periodically repeated change (~ 10 dB) of the received signal level was observed while the ship was approaching the MANNING's position, where a transducer submerged at 30 m water depth transmitted a 334 Hz CW signal. The array was towed at 45 m depth in a shallow-water area of about 80 m water depth with a slightly upward-sloping bottom. The interference wavelength is calculated using the approximation for a 'soft' bottom of $\lambda_{n,m} \approx 8 H^2 f / c_1 (n^2 - m^2)$. From a comparison of the experimental and theoretical results for the first three modes, it is believed that the first and second modes have the strongest influence on this change in level. Table 4 compares the calculated and measured interference wavelength.

TABLE 4
INTERFERENCE WAVELENGTH $\lambda_{n,m}$ FOR TWO DIFFERENT WATERDEPTHS

<u>THEORETICALLY</u>		
<u>Mode</u>	H = 70 m	H = 80 m
1+2	2909 m	3800 m
2+3	1745 m	2280 m
1+3	1091 m	1425 m

<u>OBSERVED</u>	
H = 70 m	H = 80 m
~ 3000 m	~ 3700 m

No further investigation was made during the sea trial. The observed values shown in Fig. 7 indicate the range from own ship to the transmitting site.

In the present sea-trial the water depth was generally greater than 150 m, the frequency over 100 Hz, and the cable scope of the towed array less than 750 m. Therefore straight-line ray tracing rather than normal-mode arrivals was used to estimate the angle of arrival at the array of the most likely paths <7>:

- Direct (D)
- Bottom-bounce (B)
- Bottom/surface (BS)
- Bottom/surface/bottom (BSB)
- Bottom/surface/bottom/surface (BSBS)

Figure 8 shows the Maximum Likelihood Beamformer (MLB) output and the corresponding water depths and arrival angles for (a) 160 Hz in Run 1, (b) 335.2 Hz in Run 3, and (c) 280 Hz in Run 4. The angles given on the sea-depth profile correspond to the paths quoted above.

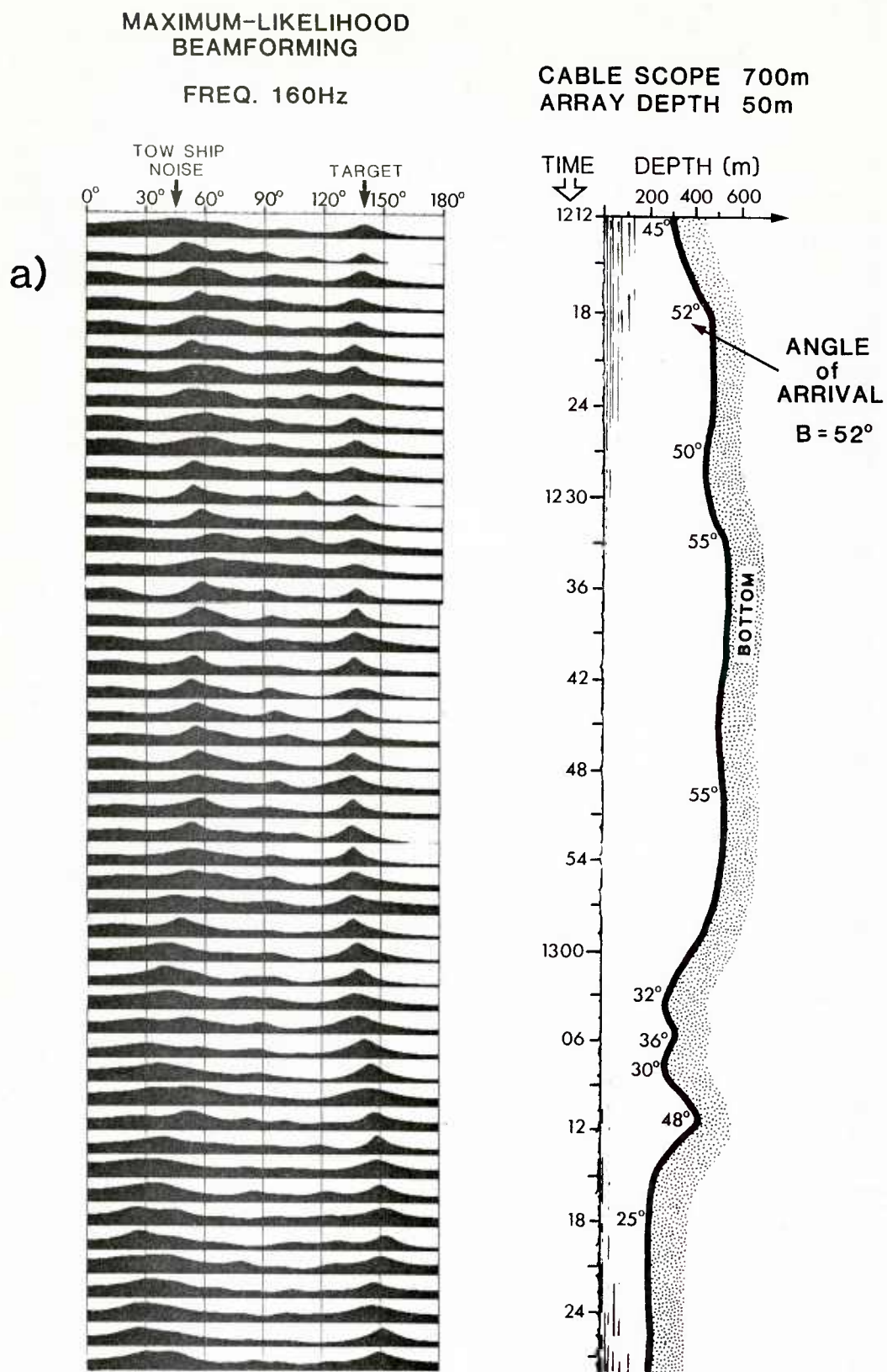


FIG. 8a MAXIMUM LIKELIHOOD BEAMFORMING OUTPUT SHOWING CORRESPONDING
WATER DEPTHS AND ARRIVAL ANGLES AT THE BOTTOM
160 Hz (Run 1)

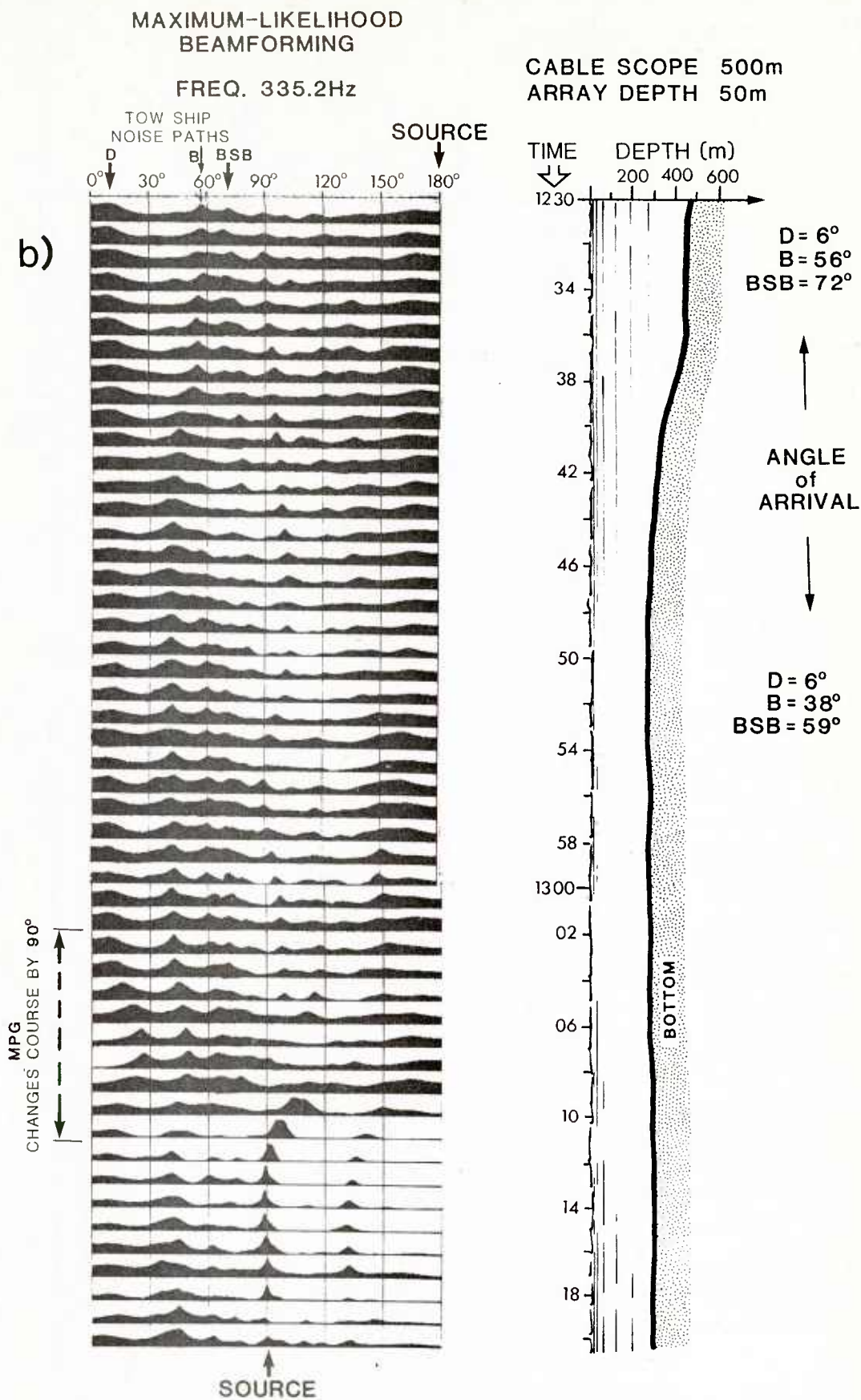


FIG. 8b MAXIMUM LIKELIHOOD BEAMFORMING OUTPUT SHOWING CORRESPONDING
WATER DEPTHS AND ARRIVAL ANGLES AT THE BOTTOM
335.2 Hz (Run 3)

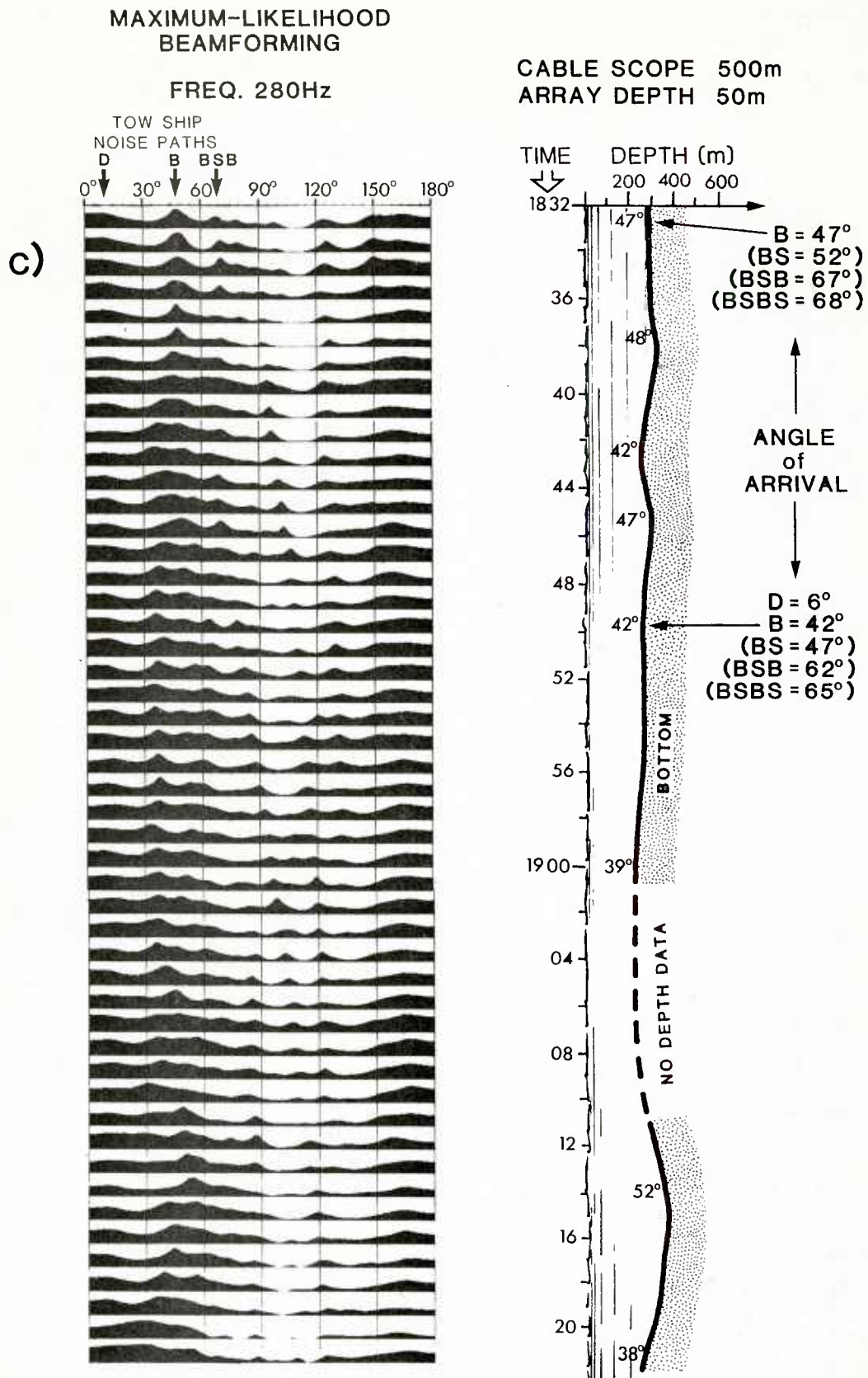


FIG. 8c MAXIMUM LIKELIHOOD BEAMFORMING OUTPUT SHOWING CORRESPONDING
WATER DEPTHS AND ARRIVAL ANGLES AT THE BOTTOM
280 Hz (Run 4)

It is seen from these figures that the bottom-bounce path (B) is usually the dominant one, although the direct path (D) and the bottom/surface/bottom path (BSB) are also observed, particularly at the higher frequency when the angular resolution is better.

Figure 8b includes data obtained whilst the towship was making a 90° turn and it is seen how the array at 500 m scope lagged behind the MPG during the turn so that the direct path made a bigger angle with the array. These direct-path data indicate how, at this frequency, when the array is looking end-fire it can indicate the vertical angle of arrival of a signal. For example, the direct path under the steady-state conditions of Run 3 arrives at about 5° (arcsin 40/450), as shown in Fig. 8b.

4.2 Towship-noise suppression

This discussion starts with some mathematical observations about the three methods implemented:

- a. Modified eigenvalues
- b. Reference beam
- c. Reference hydrophone

The question is to determine how the methods modify the covariance matrix R given by:

$$R = \sum_{n=1}^N \lambda_n E_n E_n^* \quad , \quad (\text{Eq. 29})$$

The method of modified eigenvalues, as described in Eq. 12, results in a modified matrix R' given by

$$R' = \sum_{n=1}^N \lambda'_n E_n E_n^* \quad . \quad (\text{Eq. 30})$$

which can be written as:

$$\begin{aligned} R' &= R - \sum_{n=1}^N (\lambda_n - \lambda'_n) E_n E_n^* \\ &= R - R_{T,1} \end{aligned} \quad . \quad (\text{Eq. 31})$$

The method with reference beam

$$R' = R - \frac{(RW_r)(RW_r)^*}{W_r^* R W_r} = R - \mu_2 V_2 V_2^* = R - R_{T,2} \quad (\text{Eq. 32})$$

The method with reference hydrophone

$$R' = R - \frac{Q Q^*}{r_y} = R - \mu_3 V_3 V_3^* = R - R_{T,3} \quad . \quad (\text{Eq. 33})$$

$R_{T,1}$, $R_{T,2}$ and $R_{T,3}$ respectively, are what the three methods estimate as the contribution to the covariance matrix R from the towship. According to definition (App. D), $R_{T,1}$ will have rank M if M eigenvalues are modified, while $R_{T,2}$ and $R_{T,3}$ are seen always to have rank 1 (only one non-zero eigenvalue). But it is shown in App. C that the rank is 1 only for a completely coherent reception. Hence the two last methods may work only if the real contribution $R_{T,0}$ has rank 1. The experimental results, however, indicate strongly that this is not so.

The three methods are discussed individually below. Figure 9 gives the line-printer outputs supporting the discussion, although no data are shown for the Reference Hydrophone Technique (Sect. 4.2.3).

4.2.1 The method with modified eigenvalues

If M eigenvalues are modified, then:

$$R_{T,1} = \sum_{n=1}^N (\lambda_n - \lambda'_n) E_n E_n^* \quad n \in \{n_1, n_2, \dots, n_m\} \quad . \quad (\text{Eq. 34})$$

Hence this method can easily create a modification of any rank. But that is all; almost as expected, the method is more or less useless. It was implemented simply because it was so simple to implement. The obvious reason that it does not work well is that there is not a one-to-one correspondence between eigenvalues vectors and sources. The contribution from one source is usually distributed on many eigenvalues, and one eigenvalue is a combination of contributions from several sources. The only simple case is with orthogonal sources.

Still, the method gave insight. It was quite interesting to see the "correlation" between the beam-steering vectors and the eigenvectors:

$$C_{ij} = |B_i^* E_j|^2 \quad 0 \leq C_{ij} \leq 1 \quad (\text{Eq. 35})$$

when both B and E are normalized. During the sea-trial, C_{ij} was printed out for the two eigenvectors belonging to the two largest eigenvalues (part of the tabulated data given to create Fig. 6). When signals were transmitted from the source below the workboat it was often possible to

CONVENTIONAL BEAMFORMING EFFECT OF NOISE SUPPRESSION TECHNIQUES

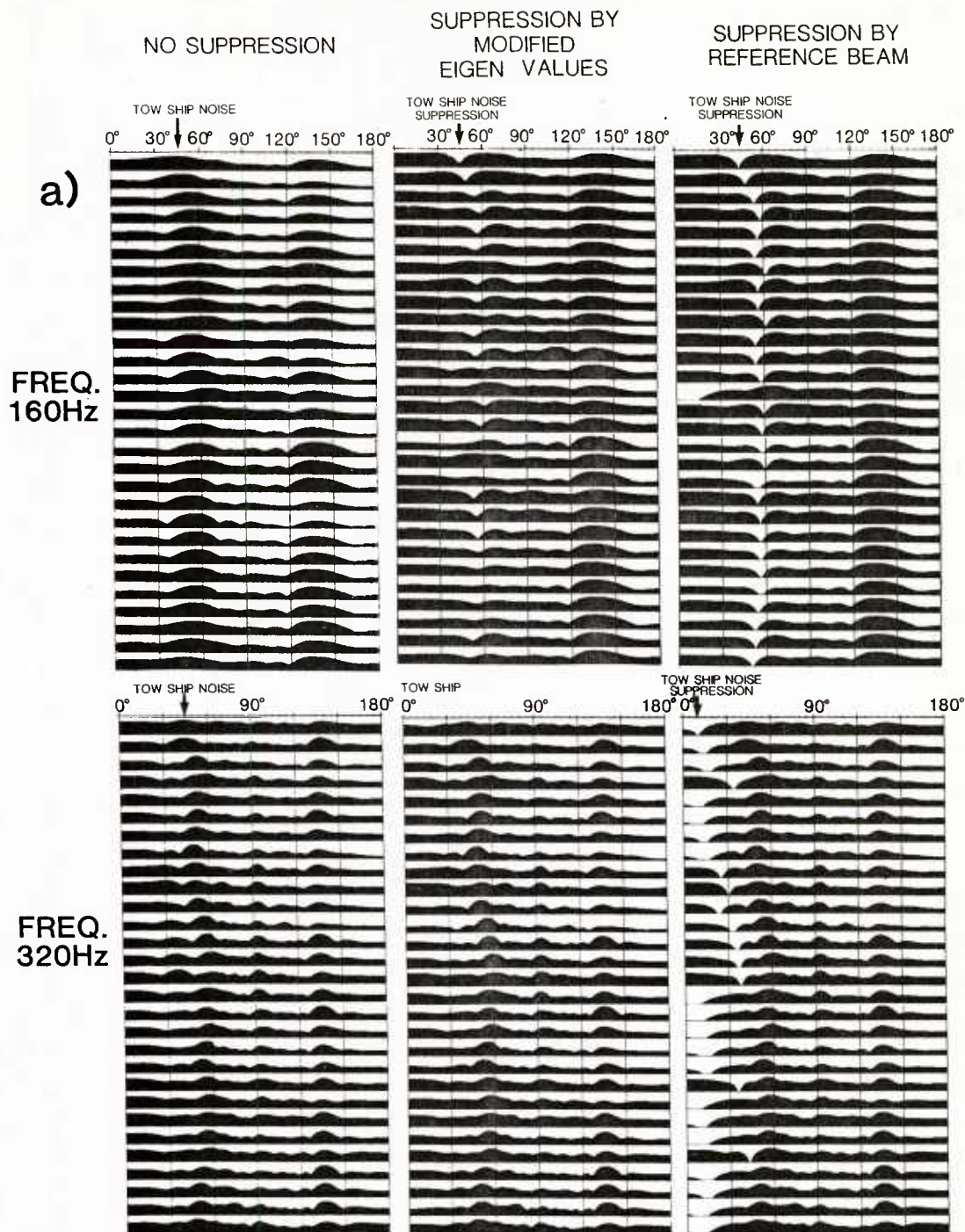


FIG. 9a COMPARISON OF NOISE SUPPRESSION TECHNIQUES
Run 1 (160 and 320 Hz) processed with conventional beamforming

MAXIMUM-LIKELIHOOD BEAMFORMING EFFECT OF NOISE SUPPRESSION TECHNIQUES

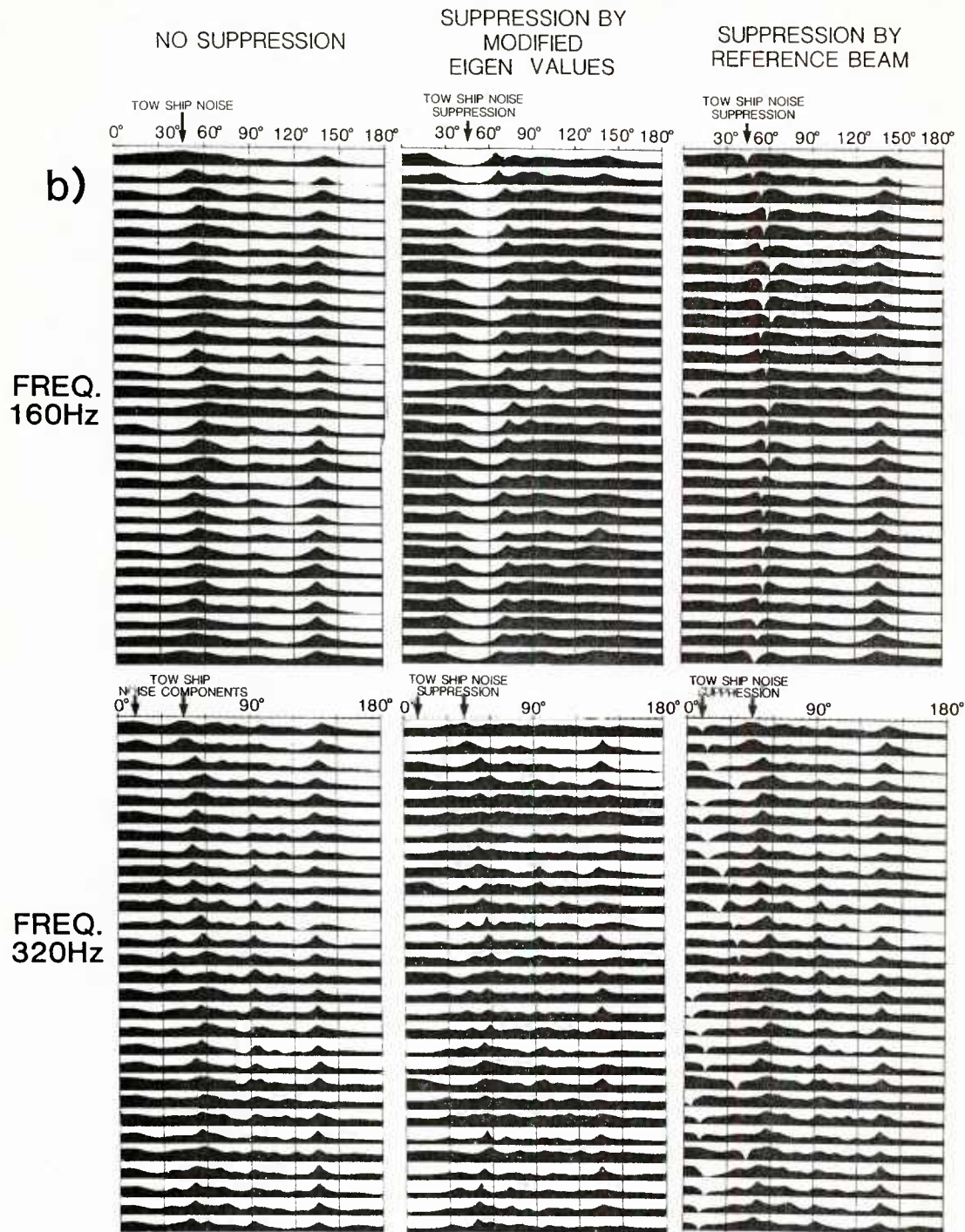


FIG. 9b *COMPARISON OF NOISE SUPPRESSION TECHNIQUES*
Run 1 (160 and 320 Hz) processed with maximum likelihood beamforming

CONVENTIONAL BEAMFORMING EFFECT OF NOISE SUPPRESSION TECHNIQUES

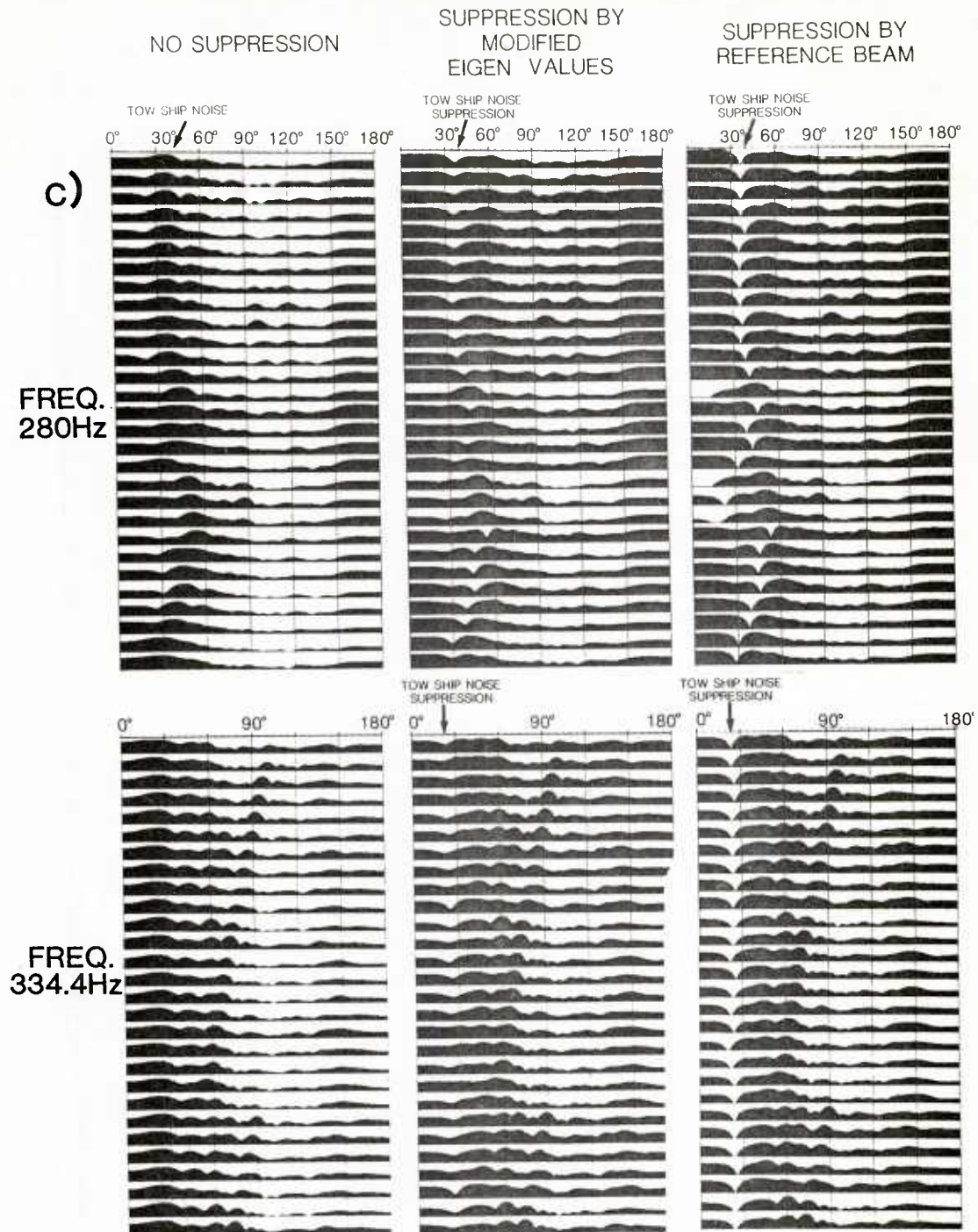


FIG. 9c *COMPARISON OF NOISE SUPPRESSION TECHNIQUES*
Run 2 (280 and 334.4 Hz) processed with conventional beamforming

**MAXIMUM-LIKELIHOOD BEAMFORMING
EFFECT OF NOISE SUPPRESSION TECHNIQUES**

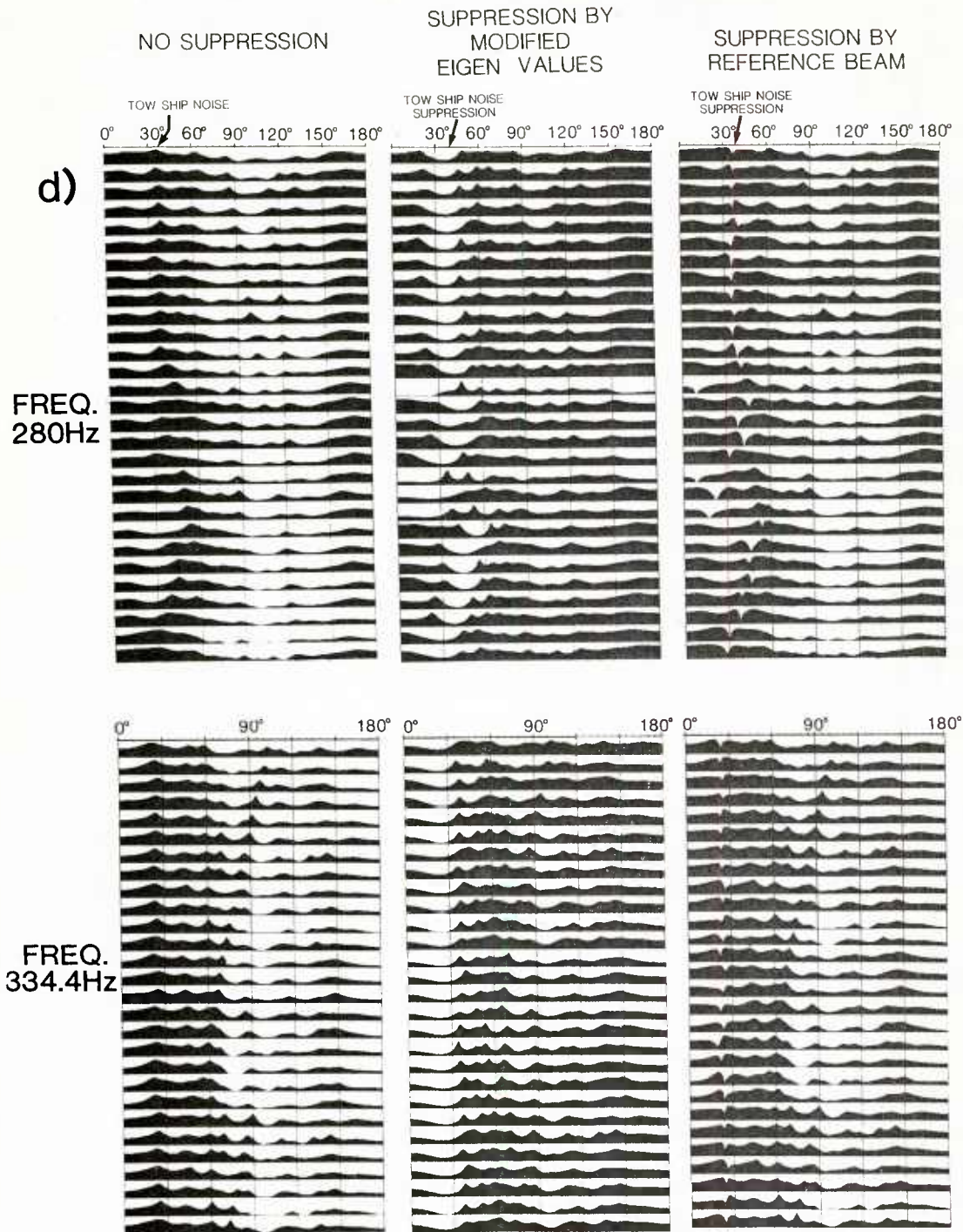


FIG. 9d *COMPARISON OF NOISE SUPPRESSION TECHNIQUES*
Run 2 (280 and 334.4 Hz) processed with maximum likelihood beamforming

observe how the two eigenvalues E_1 and E_2 were correlated with beam directions corresponding both to the towship noise and to the workboat source. This means that we could easily suppress the towship noise by modifying the first three to four eigenvalues, but at the same time the other signals would also be suppressed in an uncontrolled, random, and intolerable way.

At this point it may be appropriate to speculate why the towship noise seen by the array towed 500 m behind the ship has a covariance matrix of rank 3 to 4. There are three possibilities:

a) That the geometry of the array relative to the ship is changing during the averaging time of the covariance matrix. This time was 40 to 60 s in the sea-trials, which is obviously long compared with the possible snaking of the array. The solution would be to use a much shorter averaging time and, if possible, to overlap the FFT-windows in the filtering process before the averaging, so as to use the data efficiently.

b) That the different paths of the multipath propagation from the towship to the array have different travel times. As the continuous background spectrum of the towship has a short correlation time, the contribution at a given time from the different paths may have a reduced cross-correlation, which would mean a non-coherent signal from the towship and thus a covariance matrix of rank higher than one.

c) That the influence of the medium and its boundaries is reducing the coherence, even for an originally stable line (CW).

4.2.2 The method with reference beam

This method estimates the covariance matrix of towship noise to be

$$R_{T,2} = \frac{(RW_r)(RW_r)^*}{W_r^* RW_r} \cdot \mu_2 V_2 V_2^* , \quad (\text{Eq. 36})$$

where

$$W_r = \begin{cases} B_r & \text{for CBF} \\ (B_r^* R^{-1} B_r)^{-1} R^{-1} B_r & \text{for MLBF} \end{cases} .$$

In both cases the model assumes that the towship noise arrives as a coherent plane wave at the array. This might work well in deep water; in shallow water it definitely did not. The towship noise was certainly suppressed in a bearing interval around the reference-beam direction (given by B_r), but the rest of the towship noise was more or less unchanged.

Secondly, if there was any signal (target) in the reference direction, it would also be cancelled. The plots indicate multipath arrival at different

vertical arrival angles. Hence, even if the towship noise is coherent, it would not fit a plane-wave model. Making it even more difficult, there are strong indications (from study of eigenvalues and vectors) that the towship-noise signal is not coherent over the array.

It might be possible to work out an ad hoc algorithm by choosing more than one reference beam and cross correlating between the reference beams, but it is not obvious that the outcome would be worth the effort.

4.2.3 The method with reference hydrophone

This method estimates the covariance matrix of the towship noise to be

$$R_{T,3} = \frac{Q Q^*}{r_y} = \mu_3 V_3 V_3^* \quad , \quad (\text{Eq. 37})$$

where

$$Q = \frac{1}{M} \sum_{i=1}^M x_i y_i^*$$

and

$$r_y = \frac{1}{M} \sum_{i=1}^M |y_i|^2 \quad . \quad (\text{Eq. 38})$$

As seen, the method assumes the covariance to be of rank 1, which means a coherent signal. But it is an improvement (theoretically, at least) from the reference-beam method in that it does not assume anything about the shape of the wavefront. Furthermore, even a signal with the same directivity pattern as the towship noise should be only slightly reduced if:

$$\left(\frac{\sigma^2_{\text{ref}}}{\sigma^2_{\text{array}}} \right)_{\text{Towship}} \gg \left(\frac{\sigma^2_{\text{ref}}}{\sigma^2_{\text{array}}} \right)_{\text{Target}} \quad ,$$

where

σ^2_{ref} = Towship or Target power on the reference hydrophone

σ^2_{array} = Towship or Target power on the array hydrophone.

This is exactly the reason why the reference hydrophone has to be closer to the towship than to the array. From a theoretical point of view this should be the best of the three implemented methods. Although this sounds promising, the results so far show that the method in its present shape does not work at all. There are two possible reasons for this:

a) The first is that the reference hydrophone was rather simply designed and thus created flow noise. This would especially influence its

power estimate r_y . This estimate was therefore modified by a factor smaller than 1, to see what would happen:

$$r_y' = q_n r_y, \quad \text{where } 0 \leq q \leq 1.$$

This was done at different frequencies, including the frequency of 334 Hz transmitted by the workboat.

The result was disastrous. The workboat signal disappeared almost completely, while the towship noise was still there, reduced only for some bearings. Looking a little closer at the mathematics, let B_T be a direction where we can identify towship noise, and B_S be the direction of the workboat source. Then, in obvious notation:

$$P_T = B_T^* R B_T, \quad (\text{Eq. 39})$$

$$P_T' = B_T^* \left(R - \frac{Q Q^*}{r_y} \right) B_T = B_T^* R B_T - \frac{|B_T^* Q|^2}{r_y}, \quad (\text{Eq. 40})$$

$$\frac{P_T'}{P_T} = 1 - \frac{1}{r_y} \frac{|B_T^* Q|^2}{B_T^* R B_T}. \quad (\text{Eq. 41})$$

Similarly:

$$\frac{P_S'}{P_S} = 1 - \frac{1}{r_y} \frac{|B_S^* Q|^2}{B_S^* R B_S}. \quad (\text{Eq. 42})$$

To obtain a good suppression of the towship noise relative to the suppression of the workboat signal, we must have:

$$\frac{P_T'}{P_T} \ll \frac{P_S'}{P_S}, \quad (\text{Eq. 43})$$

which is the same as:

$$\frac{|B_T^* Q|^2}{B_T^* R B_T} \gg \frac{|B_S^* Q|^2}{B_S^* R B_S}. \quad (\text{Eq. 44})$$

This is in many (maybe most) situations, not the case.

The second possibility is that the change of the geometry relevant to towship noise (towship source, reference hydrophone, array) during the

averaging time may have been much more damaging than the changes in geometry relevant to the workboat signal. There would always have been some heaving movement of the towship that could have changed the position of the noise source and the nature of the multipath pattern of its noise propagation. The implemented averaging time was certainly too long to allow an adjustment to the dynamics.

Even if the geometry was completely frozen, the present implementation might still not work well. The reason is, as mentioned earlier, that the rank of the towship noise at the array was higher than 1, at least of the order 3 to 4 at 334 Hz. Hence it cannot be suppressed by a rank-1 matrix.

There are two ways to increase the rank of the reference system. One is to increase the number of reference hydrophones. This might help to some extent, but it will not help if the reason for reduced coherence is the different travel times along the different paths. Another possibility is to work with several delayed versions of the reference hydrophone output, to create a vector:

$$y(t) = \begin{cases} y(t-\tau_1) \\ y(t-\tau_2) \\ y(t-\tau_L) \end{cases} \quad (\text{Eq. 45})$$

By going through the mathematics, this will result in:

$$R' = R_{xx} - R_{xy} R_{yy}^{-1} R_{yx} = R - R_{T,y} \quad (\text{Eq. 46})$$

where

$$R_{xy} = \frac{1}{M} \sum_{i=1}^M X_i Y_i^* \quad , \quad \text{etc.} \quad (\text{Eq. 47})$$

The dimension L of Y should be such that R_{yy} has full rank. Otherwise it cannot be inverted, and in fact has more elements than can be used. Ideally the rank of R_{yy} should be the same as the rank of the towship-noise contribution to R.

The problems of flow noise could be overcome by creating a new reference-sensor system with reduced flow noise and low, or almost no, sensitivity to acceleration (vibration).

4.3 Target indication and bearing estimation

As with the techniques for the suppression of towship noise, only qualitative comments can be given on the performance (Fig. 10) of the two forms of processing (CBF, MLB) for providing an indication of the presence of a target of opportunity and an estimate of the target bearing for

frequencies of 112 Hz and 334.4 Hz. Figure 10b also includes the presence of a 334.4 CW source. Points to note from Figs. 10a and b are:

- a. The CBF processing does not include any shading, therefore the sidelobes are high. However, CBF is not a serious contender for providing target data.
- b. The MLB beamforming gives a sharp and precise display, although at times low-level 'targets' are suppressed, due to the maximum level used on the normalizing factor in the display.

CONCLUSIONS

a. A detailed analysis should be made of the signal paths between the towship and the array. An eigenvalue analysis indicated that the multipath propagation is reducing the coherence (Sect. 4.2). This may require the application of the FFP technique and normal-mode propagation (Sect. 4.1).

b. Because of the lack of a one-to-one correspondence between eigenvalue/vector and source, except when the sources are orthogonal, the modified eigenvalue technique for suppressing towship noise does not appear to be feasible (Ch. 3). However, the eigenvalues/vectors give useful information on certain characteristics of signals and will therefore be used in the future signal-processing software.

c. The use of a single reference beam for suppressing towship noise is not satisfactory; it assumes that the towship noise arrives at the array as a coherent plane wave and that there is no target signal in that direction (Ch. 3). It would be possible to select more than one beam and produce an ad hoc algorithm but its usefulness under the full range of operating conditions would be questionable. No further work is proposed at this stage.

d. The reference-hydrophone technique did not work because of the oversimplified nature of the present technique and the high level of flow-noise in the reference hydrophone (Ch. 3). A technique to provide a more accurate representation of the towship noise is being developed for future sea trials, and the hydrophone design is being improved to keep the flow-noise level to a minimum using a separate streamer for the reference hydrophone.

TS TOW SHIP
T TARGET

FREQUENCY 112Hz

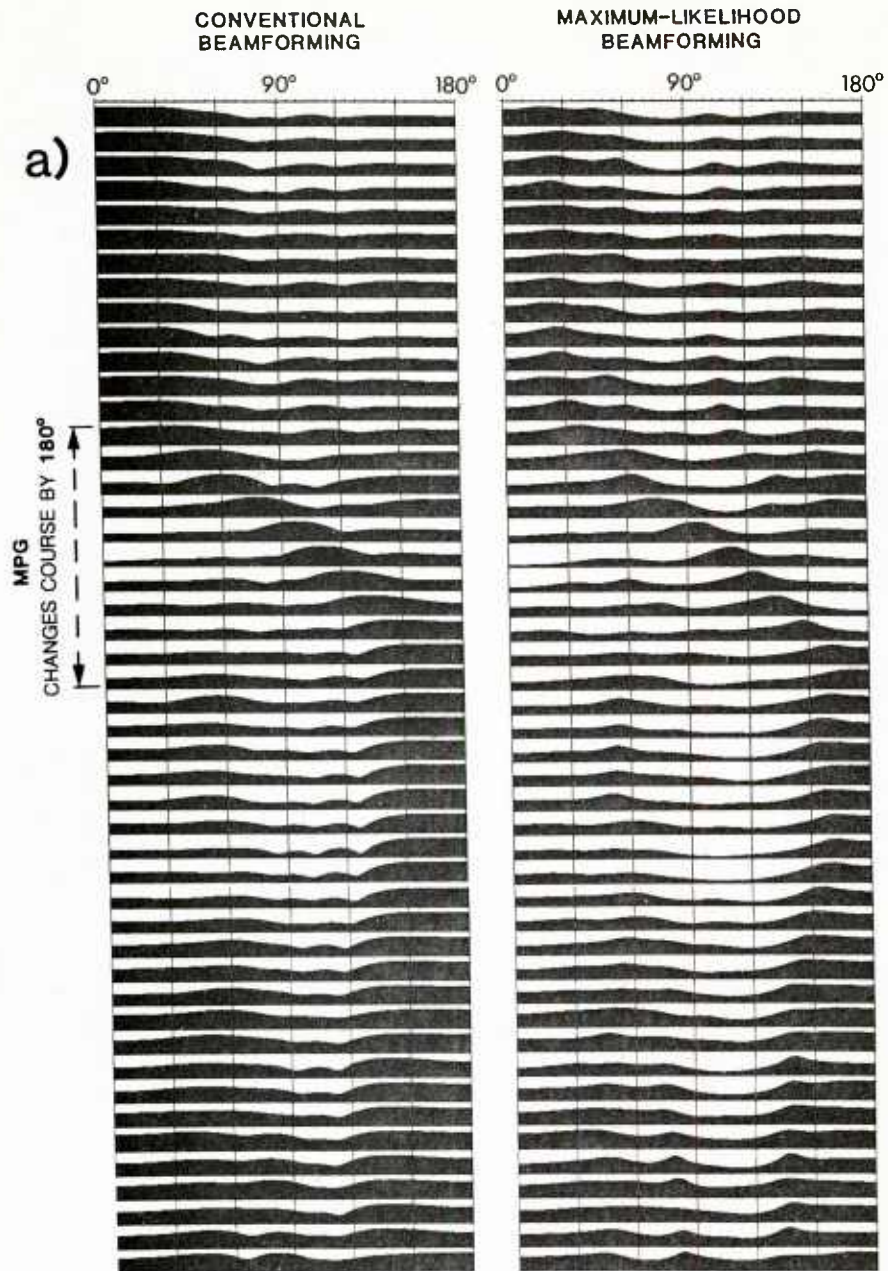


FIG. 10a COMPARISON OF TARGET-BEARING INDICATION BY CONVENTIONAL BEAMFORMING AND MAXIMUM LIKELIHOOD BEAMFORMING
Low frequency, 112 Hz, Target only

TS TOW SHIP
 T TARGET
 S SOURCE
 SL SOURCE LEVEL (dB)

FREQUENCY 334.4Hz

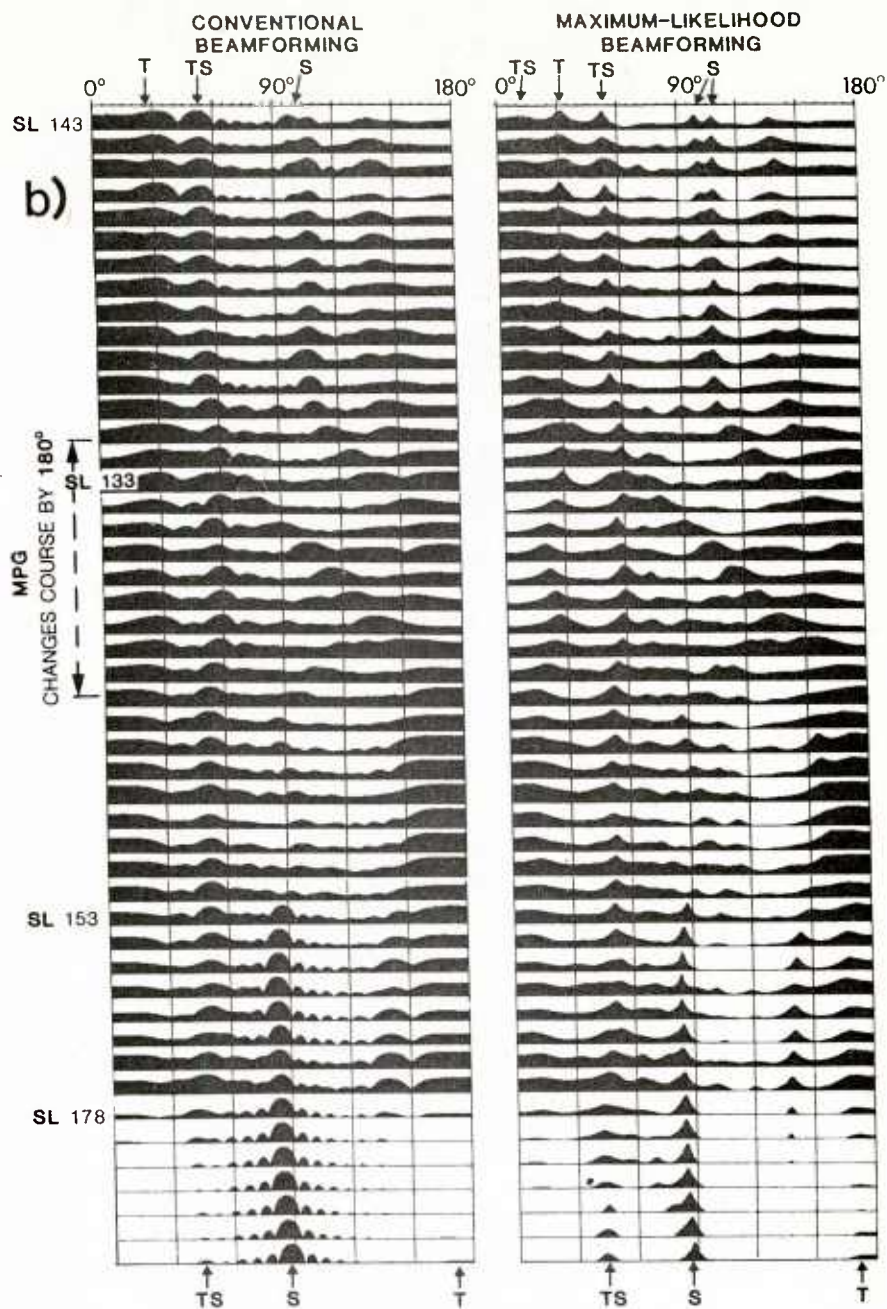


FIG. 10b COMPARISON OF TARGET-BEARING INDICATION BY CONVENTIONAL BEAMFORMING AND MAXIMUM LIKELIHOOD BEAMFORMING
 High frequency, 334.4 Hz, Target plus CW source

ACKNOWLEDGEMENTS

The following members of SACLANTCEN contributed to the work reported in this document.

<u>Signal Processing Group</u>	L. Lloyd	Assistance with first draft and criticism.
<u>Electronic/Acoustic</u>	E. Muzi	Assistance with sea trials.
<u>Engineering Department</u>	P. Saia	Assistance with sea trials.

REFERENCES

1. AKAL, T., FIORI, S., HASTRUP, O.F., and PARISOTTO, A. Acoustic transmission loss data for some selected shallow-water areas, SACLANTCEN SR-33, NATO CONFIDENTIAL. La Spezia, Italy, SACLANT ASW Research Centre, 1979. [AC C 950 788]
2. JENSEN, F.B. Comparison of transmission loss data for different shallow-water areas with theoretical results provided by a three-fluid normal-mode propagation model. In: HASTRUP, O.F. and OLESEN, O.V. eds. Sound propagation in shallow water - Vol. II: Unclassified papers. Proceedings of a Conference held at SACLANTCEN on 23-27 September 1974, SACLANTCEN CP-14. La Spezia, Italy, SACLANT ASW Research Centre, 1974: 79-92. [AD A 004 805]
3. HEWLETT-PACKARD. Vector Instruction Set (VIS). User's Manual Hewlett-Packard Computer. Libr. index No. 2 VIS. 320. 12824-90001, Part No 12824-90001. Cupertino, CA, Hewlett-Packard Company, 1979.
4. JENSEN, F.B. and KUPERMAN, W.A. Environmental acoustical modelling at SACLANTCEN, SACLANTCEN SR-34. La Spezia, Italy, SACLANT ASW Research Centre, 1979. [AD A 081 853]
5. KLEMM, R. and RUPE, U.E. An experimental study of the suppression and resolution of towing-ship noise when using towed arrays, SACLANTCEN SM-143, NATO RESTRICTED. La Spezia, Italy, SACLANT ASW Research Centre, 1980.
6. BLATSTEIN, Ira M. Comparison of normal mode theory, ray theory, and modified ray theory for arbitrary sound velocity profiles resulting in convergence zones, NOLTR 74-95. White Oak, MD, US Naval Ordnance Laboratory, 1974. [AD 787 629].
7. URICK, R.J. Principles of Underwater Sound for Engineers. New York, N.Y., Mc Graw-Hill Book Company, 1967.

APPENDICES

APPENDIX A

COVARIANCE MATRIX ESTIMATE

This appendix gives seven definitions with their associated properties and "proofs" relevant to our use of the covariance matrix estimate, R . The main emphasis is on eigenvalues and eigenvectors.

The square matrix R of dimension N is defined by:

$$R = \frac{1}{M} \sum_{i=1}^M X_i X_i^* \quad , \quad (\text{Eq. A.1})$$

where X will be called a data vector, and is of dimension N , and the star indicates a complex conjugate transpose.

A.1 DEFINITION

A square matrix P is called hermitian if it is equal to its own complex conjugate transpose, P^* .

Property 1: R is a hermitian matrix. (P1)

Proof:

$$R^* = \frac{1}{M} \left(\sum_{i=1}^M X_i X_i^* \right)^* = \frac{1}{M} \sum_{i=1}^M (X_i^*)^* (X_i)^* = \frac{1}{M} \sum_{i=1}^M X_i X_i^* = R \quad (\text{Eq. A.2})$$

Property 2 $Z = Y^* R Y$ is a real scalar for all vectors Y : (P2)

Proof:

$$Z^* = (Y^* R Y)^* = Y^* R^* Y$$

By use of (P1):

$$Z^* = Y^* R^* Y = Y^* R Y = Z \quad \text{q.e.d.}$$

A.2 DEFINITION

A square hermitian matrix P is positive definite if, for all vectors Y except the null vector, $Y^* P Y > 0$, and non-negative definite if $Y^* P Y \geq 0$.

Property 3: R is a non-negative definite matrix. (P3)

Proof:

$$\begin{aligned} Z &= Y^*RY = \frac{1}{M} \sum_{i=1}^M Y^*X_i X_i^*Y = \frac{1}{M} \sum_{i=1}^M (Y^*X_i)(Y^*X_i)^* \\ &= \frac{1}{M} \sum_{i=1}^M |Y^*X_i|^2 \geq 0 \quad \text{q.e.d.} \end{aligned} \quad (\text{Eq. A.3})$$

A.3 DEFINITION

If E is a vector, but not a null vector, such that $PE = \lambda E$, then E is called an eigenvector, and λ the corresponding eigenvalue. The eigenvectors are always assumed to be normalized: $E^*E = 1$.

Property 4: the eigenvalues of R are real and non-negative. (P4)

Proof:

$$\lambda E = RE$$

Premultiplying with E^* :

$$\lambda = \lambda E^*E = E^*RE$$

According to (P2), λ is real, and according to (P3), λ is non-negative.

A.4 DEFINITION

Two vectors X and Y , not being null-vectors, are orthogonal if $X^*Y = 0$. They are called orthonormal if both X and Y are normalized by having unit length.

Property 5: Eigenvectors of R with different eigenvalues are orthogonal.

Proof:

$$\lambda_m E_n^* E_m = E_n^* \lambda_m E_m = E_n^* R E_m$$

By exchanging indices:

$$\lambda_n E_m^* E_n = E_m^* R E_n$$

Then take the complex conjugate transpose of both sides of the last equation, and use (P1) and (P4):

$$\lambda_n E_n^* E_m = E_n^* R E_m$$

Combining the first and last equations of the proof:

$$(\lambda_m - \lambda_n) E_n^* E_m = 0$$

As the eigenvalues are assumed different $\lambda_n \neq \lambda_m$:

$$E_n^* E_m = 0 \quad \text{q.e.d.}$$

Property 6: There are N eigenvalues of R , which may not be all distinct (different from each other), and they are a solution to

$$\det(R - \lambda I) = 0$$

Proof:

$$\begin{aligned} RE &= \lambda E \\ (R - \lambda I) E &= 0 \end{aligned}$$

This can be true for E not being a null vector if, and only if, the determinant of $(R - \lambda I)$ is zero, $\det(R - \lambda I) = 0$.

This is a polynome in λ of degree N , and has N solutions, which may not be distinct. These N solutions are hence eigenvalues of R .

We know already that if eigenvalues are different, their corresponding eigenvectors are orthonormal. If there are k eigenvalues with the same value, we will state without proof that it is always possible to find exactly k orthonormal vectors that are eigenvectors of R with this same value on their eigenvalues. Hence we state without proof the following property:

Property 7: R has always N orthonormal eigenvectors. (P7)

Property 8: The eigenvectors with positive eigenvalues are linear combinations of the data vectors.

Proof:

$$\lambda E = RE = \frac{1}{M} \sum_{i=1}^M X_i X_i^* E = \frac{1}{M} \sum_{i=1}^M (X_i^* E) X_i \quad \text{q.e.d.}$$

Property 9: Eigenvectors, which are linear combinations of the data vectors, have positive eigenvalues. (P9)

Proof:

$$\lambda = \lambda E^* E = E^* R E = \frac{1}{M} \sum_{i=1}^M E^* X_i X_i^* E = \frac{1}{M} \sum_{i=1}^M |X_i^* E|^2 > 0 \quad \text{q.e.d.}$$

A.5 DEFINITION

The k vectors Y are said to be linearly independent if

$$\sum_{i=1}^K c_i Y_i = 0 \quad \text{only if } c_i = 0 \text{ for } i = 1, \dots, k$$

Property 10: If the number of independent data vectors is M , and $M < N$, $N - M$ eigenvectors are orthogonal to the data vectors, and their corresponding eigenvalues are zero. (P10)

"Proof":

From M independent data vector, we state without proof that M eigenvectors be constructed:

$$E_m = \sum_{i=1}^M c_i X_i \quad m = 1, 2, \dots, M$$

or, in a more compact form:

$$\begin{bmatrix} E_1, E_2, \dots, E_M \end{bmatrix} = \begin{bmatrix} X_1, X_2, \dots, X_M \end{bmatrix} C$$

As the M eigenvectors are also independent, C must be non-singular, and C^{-1} exists:

$$\begin{bmatrix} X_1, X_2, \dots, X_M \end{bmatrix} = \begin{bmatrix} E_1, E_2, \dots, E_M \end{bmatrix} C^{-1}$$

Premultiply with an eigenvector E_n $n = M+1, \dots, N$

$$\begin{aligned} \begin{bmatrix} E_n^* X_1, E_n^* X_2, \dots, E_n^* X_M \end{bmatrix} &= \begin{bmatrix} E_n^* E_1, E_n^* E_2, \dots, E_n^* E_M \end{bmatrix} C^{-1} \\ &= \begin{bmatrix} 0, 0, \dots, 0 \end{bmatrix} C^{-1} = 0 \end{aligned}$$

by using (P7). Hence

$$E_n^* X_i = 0 \quad \text{for } i = 1, 2, \dots, M \quad n = M+1, M+2, \dots, N \quad \text{q.e.d.}$$

The corresponding eigenvalues are:

$$\lambda_n = \lambda_n E_n^* E_n = E_n^* R E_n = \frac{1}{M} \sum_{i=1}^M |E_n^* X_i|^2 = 0 \quad \text{q.e.d.}$$

A.6 DEFINITION

A square matrix P is singular if its determinant is zero, and its inverse P^{-1} does not exist.

Property 11: If the number of independent data vectors M is less than N , R is singular. (P11)

Proof: From (P10) it follows that if $M < N$, $N - M$ eigenvalues are zero. But the eigenvalues are solutions to $\det(R - \lambda I) = 0$.

Hence for $\lambda = 0$, $\det(R) = 0$ and R is singular.

Property 12: If R has N independent data vectors, it is non-singular and its inverse R^{-1} does exist. (P12)

Proof: The proof follows directly from (P8) and (P9) for $M = N$ and these data vectors are independent.

A.7 DEFINITION

A square non-singular matrix P is unitarian if its inverse P^{-1} is equal to its complex conjugate transpose P^* .

Property 13: The square matrix $S = [E_1, E_2, \dots, E_N]$ constructed from the N eigenvectors of R , are unitarian. (P13)

Proof: Because the eigenvectors are independent, S is non-singular, and its inverse S^{-1} exists. Let us introduce:

$$\begin{aligned} T &= (S^{-1})^* \\ &= [T_1, T_2, \dots, T_N] \end{aligned}$$

where T_i is a vector.

These vectors can be expressed as linear functions of the eigenvectors:

$$T_i = \sum_{j=1}^N \alpha_{ij} E_j$$

From $T^*S = S^{-1}S = I$ it follows that:

$$T_i^* E_n = \sum_{j=1}^N \alpha_{ij}^* E_j^* E_n = \alpha_{in}^* = 1 \text{ if } i = n$$

$$= 0 \text{ if } i \neq n$$

From this follows:

$$T_i = E_i \quad \text{and} \quad T = S = (S^{-1})^* \quad \text{q.e.d.}$$

Property 14: R can be written as $R = SD S^*$, where $D = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ is a diagonal matrix. (P14)

Proof:

$$RE_i = \lambda_i E_i \quad i = 1, 2, \dots, N \quad \text{can be combined to:}$$

$$RS = SD$$

As S is non-singular:

$$R = S D S^{-1}$$

and from (P13):

$$R = S D S^* \quad \text{q.e.d.}$$

Property 15: If R is non-singular, its inverse can be written:

$$R^{-1} = S D^{-1} S^* \quad . \quad (P15)$$

Proof:

$$R^{-1} = (S D S^*)^{-1} = (S^*)^{-1} D^{-1} S^{-1}$$

Using (P13), this is

$$R^{-1} = S D^{-1} S^* \quad \text{q.e.d.}$$

Property 16: Let n be an integer, then

$$R^n = S D^n S^*$$

Proof:

For $n \geq 1$

$$R^n = (SDS^*) (SDS^*) \dots (SDS^*) = SDS^{-1} SDS^{-1} \dots SDS^* = SD^n S^* \quad \text{q.e.d.}$$

For $n \leq -1$

$$R^{-n} = (SD^{-1} S^*)(SD^{-1} S^*) \dots (SD^{-1} S^*) = SD^{-1} S^{-1} S D^{-1} S^{-1} \dots SD^{-1} = SD^{-n} S^* \quad \text{q.e.d.}$$

For $n = 0$

$$R^0 = I = SS^{-1} = SS^* = SD^0 S^* \quad \text{q.e.d.}$$

Property 17: Let n be an integer, then

$$R^n = \sum_{i=1}^N \lambda_i^n E_i E_i^*$$

Proof:

$$D^n = [\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)]^n = \text{diag}(\lambda_1^n, \lambda_2^n, \dots, \lambda_N^n)$$

by direct use of matrix multiplication. Hence:

$$\begin{aligned}
 R^n &= [E_1, E_2, \dots, E_N] \begin{bmatrix} \lambda_1^n & & & \\ & \lambda_2^n & & \\ & & \ddots & \\ & & & \lambda_N^n \end{bmatrix} \begin{bmatrix} E_1^* \\ E_2^* \\ \vdots \\ E_N^* \end{bmatrix} \\
 &= [E_1, E_2, \dots, E_N] \begin{bmatrix} \lambda_1^n E_1^* \\ \lambda_2^n E_2^* \\ \vdots \\ \lambda_N^n E_N^* \end{bmatrix} = \sum_{i=1}^N \lambda_i^n E_i E_i^* \quad \text{q.e.d.}
 \end{aligned}$$

Property 18: For all X and Y :

$$X^* R^n Y = \sum_{i=1}^N \lambda_i^n (X^* E_i) (E_i^* Y)$$

Proof: This follows directly from (P17).

APPENDIX BBEAMFORMING AND NOISE SUPPRESSIONB.1 CONVENTIONAL BEAMFORMING (CB)

Let X_i be a narrowband complex data vector of dimension N (N array hydrophones) from a frequency band around f , where i is a time index. Then the covariance matrix estimate R is defined by:

$$R = \frac{1}{M} \sum_{i=1}^M X_i X_i^* \quad M \geq N, \quad (\text{Eq. B.1})$$

where $1/M$ is added for convenience.

A steering vector B is defined by:

$$B^T = \left[1, \dots, e^{-j2\pi \frac{x_n}{\lambda} \cos \beta} \right] \frac{1}{\sqrt{N}},$$

where x_n is the position of the n -th hydrophone, β is the bearing angle measured from ahead endfire, and \sqrt{N} is introduced to normalize B :
 $B^* B = 1$.

A "conventional" beam is then:

$$Z_{k,i} = B_k^* X_i, \text{ where } i \text{ and } k \text{ are time and bearing indices.}$$

The power output is:

$$P_{k,i} = |Z_{k,i}|^2 = Z_{k,i} Z_{k,i}^* = B_k^* X_i X_i^* B_k.$$

The mean (in time) conventional beam power output is:

$$P_k = \frac{1}{M} \sum_{i=1}^M P_{k,i} = \frac{1}{M} B_k^* \sum_{i=1}^M X_i X_i^* B_k = B_k^* R B_k.$$

Using eigenvalues and eigenvectors, this can be written:

$$P_k = \sum_{n=1}^N \lambda_n (B_k^* E_n) (E_n^* B_k) = \sum_{n=1}^N \lambda_n |B_k^* E_n|^2$$

B.2 MAXIMUM LIKELIHOOD BEAMFORMING (MLB)

Instead of the steering vector B , use a weight vector W to form a beam:

$$Z_{k,i} = W_k^* X_i.$$

The so-called maximum-likelihood beamforming is obtained by choosing a weight vector W that is minimizing

$$W_k^* R W_k \quad \text{constrained by} \quad W_k^* B_k = B_k^* W_k = 1$$

$$Y_k = W_k^* R W_k = (W_k^* - W_0^*) R (W_k - W_0) + W_k^* R W_0 + W_0^* R W_k - W_0^* R W_0$$

The constraint can now be introduced by choosing

$$Y_k = (W_k - c R^{-1} B_k)^* R (W_k - c R^{-1} B_k) + c + c^* - c c^* B_k^* R^{-1} B_k$$

Assuming R to be positive definite, this has a minimum for:

$$W_k = c R^{-1} B_k$$

The constant c is determined from the constraint

$$B_k^* W_k = c B_k^* R^{-1} B_k = 1$$

$$W_k = (B_k^* R^{-1} B_k)^{-1} R^{-1} B_k$$

The ML beamformer output will be:

$$Z_{k,i} = (B_k^* R^{-1} B_k)^{-1} (B_k^* R^{-1} X_i)$$

and the mean power output will be:

$$\begin{aligned} P_k &= W_k^* R W_k = (B_k^* R^{-1} B_k)^{-1} (B_k^* R^{-1} R R^{-1} B_k) (B_k^* R^{-1} B_k) \\ &= (B_k^* R^{-1} B_k)^{-1} \end{aligned}$$

Using eigenvectors and eigenvalues of R :

$$P_k = \left[\sum_{n=1}^N \lambda_n^{-1} (B_k^* E_n) (E_n^* B_k) \right]^{-1} = \left(\sum_{n=1}^N \lambda_n^{-1} |B_k^* E_n|^2 \right)^{-1}$$

B.3 NOISE SUPPRESSION WITH REFERENCE BEAM

Let there be a reference beam:

$$Y_i = W_r^* X_i \quad \text{where } W_r = B_r \quad \text{for CBF}$$

$$= (B_r^* R^{-1} B_r)^{-1} R^{-1} B_r \quad \text{for MLBF}$$

This reference beam is used to correct the beams:

$$Z'_{k,i} = Z_{k,i} - V_k Y_i$$

V_k is now chosen to minimize:

$$P'_k = \frac{1}{M} \sum_{i=1}^M |Z'_{k,i}|^2 = \frac{1}{M} \sum_{i=1}^M (Z_{k,i} - V_k Y_i) (Z_{k,i} - V_k Y_i)^*$$

$$= \frac{1}{M} \sum_{i=1}^M (Z_{k,i} Z_{k,i}^* - Z_{k,i} Y_i^* V_k^* - V_k Y_i Z_{k,i}^* + V_k V_k^* Y_i Y_i^*)$$

$$= r_z - r_{zy} V_k^* - V_k r_{zy}^* + V_k V_k^* r_y$$

with obvious notation. This can be written:

$$P'_k = \left(V_k - \frac{r_{zy}}{r_y} \right) \left(V_k - \frac{r_{zy}}{r_y} \right)^* r_y + r_z - \frac{r_{zy} r_{zy}^*}{r_y}$$

This has a minimum for:

$$V_k = \frac{r_{zy}}{r_y} = \frac{W_k^* R W_r}{W_r^* R W_r}$$

where W_k is defined similar to W_r for CBF and MLBF.

The corrected linear and mean power outputs are:

$$Z'_{k,i} = W_k^* X_i - \frac{W_k^* R W_r}{W_r^* R W_r} W_r^* X_i = \left(W_k^* - \frac{W_k^* R W_r}{W_r^* R W_r} W_r^* \right) X_i$$

$$P'_k = W_k^* R W_k - \left(W_r^* R W_r \right)^{-1} \left| W_k^* R W_r \right|^2$$

for CBF:

$$P'_k = B_k^* R B_k - (B_r^* R B_r)^{-1} \left| B_k^* R B_r \right|^2$$

Using eigenvalues and eigenvectors:

$$B_k^* R B_k = \sum_{n=1}^N \lambda_n \left| B_k^* E_n \right|^2$$

$$B_r^* R B_r = \sum_{n=1}^N \lambda_n \left| B_r^* E_n \right|^2$$

$$\left| B_k^* R B_r \right| = \sum_{n=1}^N \lambda_n (B_k^* E_n)(B_r^* E_n)^*$$

For MLBF:

$$P'_k = (B_k^* R^{-1} B_k)^{-1} - (B_r^* R^{-1} B_r)^{-1} (B_k^* R^{-1} B_r)^{-2} \left| B_k^* R^{-1} B_r \right|^2$$

And with eigenvalues and eigenvectors:

$$B_k^* R^{-1} B_k = \sum_{n=1}^N \lambda_n^{-1} \left| B_k^* E_n \right|^2$$

$$B_r^* R^{-1} B_r = \sum_{n=1}^N \lambda_n^{-1} \left| B_r^* E_n \right|^2$$

$$B_k^* R^{-1} B_r = \left| \sum_{n=1}^N \lambda_n^{-1} (B_k^* E_n)(B_r^* E_n)^* \right|$$

B.4 NOISE SUPPRESSION WITH REFERENCE HYDROPHONE

Let there be a reference hydrophone with output y . This reference hydrophone output is used to correct the array hydrophone outputs:

$$X'_i = X_i - Wy_i$$

W is now chosen to minimize:

$$p' = \frac{1}{M} \sum_{i=1}^M \left| X_i - Wy_i \right|^2 = \frac{1}{M} \sum_{i=1}^M (X_i - Wy_i)^* (X_i - Wy_i)$$

$$\begin{aligned}
&= \frac{1}{M} \operatorname{tr} \left[\sum_{i=1}^M (X_i - W y_i)(X_i - W y_i)^* \right] \\
&= \frac{1}{M} \operatorname{tr} \left[\sum_{i=1}^M (X_i X_i^* - X_i y_i^* W^* - W y_i X_i^* + W W^* y_i y_i^*) \right] \\
&= \operatorname{tr} (R - QW^* - WQ^* + WW^* r_y)
\end{aligned}$$

with obvious notation. This can be written:

$$p' = \operatorname{tr} \left[\left(W - \frac{Q}{r_y} \right) \left(W - \frac{Q}{r_y} \right)^* r_y + R - \frac{QQ^*}{r_y} \right]$$

This has a minimum for:

$$W = \frac{Q}{r_y}$$

The corrected data vector and covariance matrix will be:

$$X_i' = X_i - \frac{Q}{r_y} y_i$$

$$R' = \frac{1}{M} \sum_{i=1}^M X_i' X_i'^* = R - \frac{QQ^*}{r_y}$$

The new inverse matrix will be using the matrix inversion lemma:

$$R'^{-1} = \left(R - \frac{QQ^*}{r_y} \right)^{-1} = R^{-1} + (r_y - Q^* R^{-1} Q)^{-1} (R^{-1} QQ^* R^{-1})$$

The mean output power using CBF:

$$p_k' = B_k^* R' B_k = B_k^* R B_k - r_y^{-1} \left| B_k^* Q \right|^2$$

Using eigenvalues and eigenvectors:

$$B_k^* R B_k = \sum_{n=1}^N \lambda_n \left| B_k^* E_n \right|^2$$

$$\left| B_k^* Q \right| = \sum_{n=1}^N \lambda_n^{-1} (B_k^* E_n) Q^* E_n$$

This way of writing B^*Q , as $B^*IQ = B^*R^0Q$ may seem artificial, but may be useful from a software point of view.

The mean output power using MLBF:

$$p_k' = (B_k^* R^{-1} B_k)^{-1} = \left[B_k^* R^{-1} B_k + (r_y - Q^* R^{-1} Q)^{-1} \left| B_k^* R^{-1} Q \right|^2 \right]^{-1}$$

Using eigenvectors and eigenvalues:

$$B_k^* R^{-1} B_k = \sum_{n=1}^N \lambda_n^{-1} \left| B_k^* E_n \right|^2$$

$$Q^* R^{-1} Q = \sum_{n=1}^N \lambda_n^{-1} \left| Q^* E_n \right|^2$$

$$B_k^* R^{-1} Q = \left| \sum_{n=1}^N \lambda_n^{-1} (B_k^* E_n) (Q^* E_n)^* \right|$$

APPENDIX CTHE RANK OF A SOURCE'S CONTRIBUTION TO THE COVARIANCE MATRIX

From Appendix A, we know that the covariance matrix R can be written:

$$R = SAS^* = \sum_{i=1}^N \lambda_i E_i E_i^*$$

If only the first M eigenvalues are non-zero, R is singular, and is said to have rank M :

$$R = \sum_{i=1}^M \lambda_i E_i E_i^*$$

Let us look at the contribution from one source. If its contribution is completely coherent over the total array, then:

$$E\{x_i x_j^*\} = (E\{|x_i|^2\} \cdot E\{|x_j|^2\})^{1/2} e^{i(\psi_i - \psi_j)} = v_i v_j^*$$

leading to:

$$R_S = E\{XX^*\} = \sigma_S^2 VV^*$$

where V is a normalized vector ($V^*V = 1$). Now:

$$R_S E = \sigma_S^2 V(V^*E) \begin{cases} = \sigma_S^2 V & \text{if } E^*V = 1 \quad (E = V) \\ = 0 & \text{if } E^*V = 0 \quad (E \perp V) \end{cases}$$

This means that there is only one non-zero eigenvalue $\lambda_1 = \sigma_S^2$ with corresponding eigenvector $E_1 = V$ and the rank of R_S is 1.

If the contribution from the source is completely incoherent, then:

$$E\{x_i x_j^*\} \begin{cases} = \sigma_i^2 & \text{if } i = j \\ = 0 & \text{if } i \neq j \end{cases}$$

Then:

$$R_S = \text{diag} \{\sigma_1^2 \dots \sigma_i^2 \dots \sigma_N^2\}$$

As $\det R_S = \prod_{i=1}^N \sigma_i^2 > 0$, this matrix has always full rank, or rank equal to N .

Hence a real source will give a contribution to the matrix with a rank from 1 in the case of complete coherence, and increasing rank with decreasing coherence. We will define the field from the source to have rank M if:

$$10 \lg \left(\frac{\lambda_i}{\lambda_1} \right) \geq a \quad \text{for } i \leq M$$

$$< a \quad \text{for } i > M$$

where a is a chosen threshold (as an example, -10 dB) and the eigenvalues are arranged in decreasing order.

APPENDIX D

RAY THEORY AND NORMAL-MODE THEORY

The acoustic propagation is usually expressed in terms of normal modes for the low-frequency propagation or in terms of rays for the high-frequency range. Ray theory and normal-mode theory is briefly reviewed here. A numerical comparison and overview is in reference <6>.

D.1 RAY THEORY

The limiting form of the exact wave theory for $\lambda \rightarrow 0$, i.e. $f \rightarrow \infty$ is called ray theory and the propagation is found to be frequency-independent for this limit. To achieve the limit the wave equation for the sound pressure

$$\nabla^2 p - c(z)^{-2} \frac{\partial p}{\partial t} = 0 \quad (\text{Eq. D.1})$$

yields for $p \sim \exp(-i\omega t)$

$$\nabla^2 p + k(z)^2 p = 0 \quad (\text{Eq. D.2})$$

where $k(z)$ is the wavenumber.

The "eikonal-solution"

$$p = A(r,z) \exp[i S(r,z)] \quad (\text{Eq. D.3})$$

permits for slowly varying $A(r,z)$ the "eikonal equation"

$$(\vec{\nabla} S)^2 = k(z)^2 \quad (\text{Eq. D.4})$$

For the derived solution for p the phase function $S(r,z)$ permits the surfaces of constant phase, and

$$S(r,z) = \text{const.} \quad (\text{Eq. D.5})$$

are the wavefronts of the sound field. The vector $\vec{\nabla} S$ are normal to the wavefront and chosen to form tangents to space curves, the so-called "rays".

The path element along a ray in range and depth coordinates (r, z) is given by:

$$d\ell = \sqrt{1 + (dr/dz)^2} dz \quad (\text{Eq. D.6})$$

and the angle θ which the ray makes with the horizontal can be found as

$$\frac{dz}{d\ell} = \sin \theta ; \quad \frac{dr}{d\ell} = \cos \theta ; \quad \frac{dr}{dz} = \cot \theta \quad (\text{Eq. D.7})$$

The unit vector pointing normal to the surface $S(r, z) = \text{const.}$ and being parallel to the vector $\vec{\nabla} S$ is defined

$$\vec{n} = (dr/d\ell, dz/d\ell) \quad (\text{Eq. D.8})$$

Together with Eq. D.4 follows

$$k \frac{dr}{d\ell} = \frac{\partial S}{\partial r} ; \quad k \frac{dz}{d\ell} = \frac{\partial S}{\partial z} \quad (\text{Eq. D.9})$$

with the operator $(d/d\ell)$

$$\frac{d}{d\ell} \left(k \frac{dr}{d\ell} \right) = \frac{\partial}{\partial r} \frac{dS}{d\ell} \quad (\text{Eq. D.10})$$

and with

$$\frac{dS}{d\ell} = \frac{\partial S}{\partial r} \frac{dr}{d\ell} + \frac{\partial S}{\partial z} \frac{dz}{d\ell}$$

one has by using Eq. D.9

$$\frac{dS}{d\ell} = \left(k \frac{dr}{d\ell} \right)^2 + k \left(\frac{dz}{d\ell} \right)^2 = k n^2 \quad (\text{Eq. D.11})$$

with Eq. D.10 again

$$\frac{d}{d\ell} \left(k \frac{dr}{d\ell} \right) = \frac{d}{dr} \left(k \cos \theta \right) = 0 \quad (\text{Eq. D.12})$$

Going along $d\ell$, i.e. a ray, one has

$$k(z) \cos \theta = \text{const. or the Snell's law} \quad (\text{Eq. D.13})$$

$$\cos \theta / c(z) = \cos \theta_0 / c_0 = \text{const.} \quad (\text{Eq. D.14})$$

This law determines the variation of grazing angle θ (the angle relative to the horizontal) along the ray corresponding to the variation of sound

speed with depth z , and relating it to the corresponding quantities θ_0 , C_0 at the source. From this and Eqs. D.6, D.7 the ray equation is derived

$$r(z) = \int_{z_0}^z \frac{dz}{\sqrt{C_0/C(z) \cdot \cos^2 \theta - 1}} \quad (\text{Eq. D.15})$$

for a ray coming from the source at a grazing angle θ_0 . Solving this equation for a constant-gradient profile, yields:

$$r(z) = \int_{z_0}^z \frac{z \, dz}{\sqrt{\alpha^2 - z^2}} = -\sqrt{\alpha^2 - z^2} \quad (\text{Eq. D.16})$$

with

$$c(z) = bz$$

and

$$\alpha = \frac{\cos \theta_0}{b C_0}$$

This permits

$$\alpha^2 = r^2 + z^2 \quad (\text{Eq. D.17})$$

with proper choice of integration constants

Hence for a small gradient, the "rays" are circles with large radius, for a large gradient they are circles with small radius and for isovelocity profile the rays become straight lines. This concept can explain the sound channeling around the SOFAR channel axis.

Effects of boundary occur, when upward curving rays are reflected by the sea surface and downward rays are reflected by the sea bottom. Bottom reflections are treated by this concept assuming equal grazing angles of incidence and reflections, but with reduced amplitudes after reflection. The sea surface is almost a perfect reflector, for a low frequency (< 100 Hz), but the phase is changed by π after the reflection. If a ray has touched a caustic, which is a curve forming a geometrical envelope of rays, the phase is changed by $\pi/2$. In fact modified ray theory does consider the caustic correction.

In calculating the phase explicitly between source and receiver along the rays, Eq. D.4 yields by integrating over a ray-path.

$$S = \int k \, d\ell = \omega \int \frac{d\ell}{c} = \omega f dt = \omega T \quad (\text{Eq. D.18})$$

where T is the travel time along the ray. Each ray amplitude has to have

a phase factor $\exp(i\omega T)$ for coherent summation. In writing Eq. D.18 explicitly

$$\begin{aligned}\omega T &= \omega \int \frac{dz}{c \sin \theta} = \omega \int \frac{dz}{c \sin \theta} (\sin^2 \theta + \cos^2 \theta) \\ &= \omega \int \frac{\sin \theta}{c} dz + \omega \int \frac{\cos^2 \theta}{c \sin \theta} dz \\ &= I_1 + I_2\end{aligned}\quad (\text{Eq. D.19})$$

$$I_1 = \omega \int \frac{dz}{c} \sqrt{1 - \cos^2 \theta} \quad (\text{Eq. D.19a})$$

and with

$$\cos \theta_0 = n \cos \theta$$

$$n = c_0/c(z)$$

$$k_0 = \omega/c_0$$

we have

$$I_1 = k_0 \int \sqrt{n^2 - \cos^2 \theta_0} dz \quad (\text{Eq. D.19b})$$

For the second integral

$$I_2 = \omega \int \frac{dz}{c \sin \theta} \cos^2 \theta \quad (\text{Eq. D.20a})$$

follows with, Eq. D.7, $dz/d\ell = \sin \theta$

$$I_2 = \omega \int \frac{\cos^2 \theta}{c} d\ell \quad (\text{Eq. D.20b})$$

With the horizontal component $k_r = k \cos \theta$ of the wavenumber $k = \omega/c$ one finds from Snell's law the horizontal component

$$k \cos \theta = k_0 \cos \theta_0 = \text{const.}$$

being range-independent, i.e. a constant along the ray. This yields for Eq. D.20b)

$$I_3 = \int k_r dr = (k_0 \cos \theta_0) r \quad (\text{Eq. D.20c})$$

The total phase along the ray can be expressed

$$\begin{aligned}\omega T &= (k_0 \cos \theta_0) r + k_0 \int_{z_0}^z \sqrt{n^2 - \cos^2 \theta_0} dz \\ &= \int k_r dr + \int k_z dz\end{aligned}\quad (\text{Eq. D.21})$$

where the vertical wavenumber

$$k_z = k \sin \theta = \frac{\omega}{c_0} \sqrt{n^2 - \cos^2 \theta}$$

is depth dependent through $n = c_0/c(z)$

From this the well known relation

$$k^2 = k_r^2 + k_z^2 \quad (\text{Eq. D.22})$$

is found.

D.2 NORMAL MODE

With a monopole source at $r = 0$ and $z = z_0$ the normal mode solution for the wave equation is

$$\nabla^2 p + kp = - \frac{\delta(r) \delta(z-z_0)}{2\pi r} \quad (\text{Eq. D.23})$$

with separation of variables, the solution is of the form

$$p(r, z) = \sum_n R_n(r) D_n(z) \quad (\text{Eq. D.24})$$

with some constants a_n and k_n we can take the range function

$$R_n(r) = a_n \frac{i}{4} H_0^{(1)}(k_n r) \quad (\text{Eq. D.25})$$

which is the Green's function of the wave equation, satisfying

$$\frac{1}{r} \frac{d}{dr} \left(r \frac{dR_n}{dr} \right) + k_n^2 R_n = - \frac{\delta(r)}{2\pi r} a_n \quad (\text{Eq. D.26})$$

and for cylindrical symmetry one has

$$\nabla^2 p = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial p}{\partial r} \right) + \frac{\partial^2 p}{\partial z^2} \quad (\text{Eq. D.27})$$

Inserting Eqs. D.25 and D.26 in Eqs. D.23 and D.24 yields

$$\begin{aligned} \sum_n \left[-k_n^2 R_n D_n + R_n \frac{d^2 D_n}{dz^2} + \frac{\omega^2}{c^2} R_n D_n \right] \\ = \frac{\delta(r)}{2\pi r} \sum_n a_n D_n(z) - \frac{\delta(r) \delta(z-z_0)}{2\pi r} \end{aligned} \quad (\text{Eq. D.28})$$

The right hand side vanishes for

$$a_n = D_n(z_0) \quad (\text{Eq. D.29})$$

and using for depth-function $D_n(z)$

$$\sum_n D_n(z) D_n(z_0) = \delta(z-z_0) \quad (\text{Eq. D.30})$$

This function satisfies the depth equation.

$$\frac{d^2 D_n}{dz^2} + \left(\frac{\omega^2}{c(z)} - k_n^2 \right) D_n = 0 \quad (\text{Eq. D.31})$$

This "eigenvalue equation" permits solutions for its characteristic "eigenvalues" k_n under the constraint that

$$D_n(z_s) = D_n(z_\infty) = 0 \quad (\text{Eq. D.32})$$

The eigenvalue k_n forms a "discrete spectrum" or a continuum.

The solution for an outgoing wave is now obtained from Eq.D.24.

$$p(r,z) = e^{-i\omega t} \frac{i}{4} \sum_{n=1}^{\infty} H_0^{(1)}(k_n r) D_n(z_0) D_n(z), \quad (\text{Eq. D.33})$$

known as normal mode series. At large ranges $r \rightarrow \infty$

$$H_0^{(1)}(k_n r) \rightarrow \sqrt{\frac{2}{\pi k_n r}} e^{i(k_n r - \frac{\pi}{4})} \quad (\text{Eq. D.34})$$

corresponds to cylindrical waves, showing that the horizontal wave-numbers are constant, as they are in ray theory, too. In other words, they are range-independent. The ray-mode equivalence in terms of an "equivalent ray angle" may help interpretations. The horizontal wave number k_n for the mode solution is assumed to be equal to the horizontal wave number $k_r = \omega \cos \theta / c$ in ray theory.

Hence the n^{th} normal mode might be associated with an "equivalent ray angle θ " of that mode by

$$\theta_n(z) = \cos^{-1} \frac{k_n c(z)}{\omega}, \quad (\text{Eq. D.35})$$

showing the difference that at given depth z , arrivals in terms of rays are continuous and in terms of modes are a discrete set of arrivals. Introducing a vertical wavenumber one obtains a depth equation,

$$\frac{d^2}{dz^2} D_n(z) + k_v^2 D_n(z) = 0 \quad (\text{Eq. D.36})$$

which can be solved analytically for certain profiles.

The Eq. D.31 might be rewritten by introducing the "potential function"

$$V(z) = k_0^2 - k^2(z) \quad (\text{Eq. D.37})$$

with $k_0 = \omega/c_0$ at channel axis and the new eigenvalue

$$E_n = k_0^2 - k_n^2 \quad (\text{Eq. D.38})$$

the form of the depth equation is found

$$\frac{d^2 D_n(z)}{dz^2} + [E_n - V(z)] D_n(z) = 0 \quad (\text{Eq. D.39})$$

Hence for a given sound speed profile a related potential function exists. The infinity value of this potential determines the boundary between continuous and discrete mode propagation and is given by

$$V(\infty) = \omega^2 \left(\frac{1}{c_0^2} - \frac{1}{c_\infty^2} \right) \quad (\text{Eq. D.40})$$

The turning point for mode functions occurs if $E_n = V(z)$, whereas beyond this depth z the vertical wavenumber is negative and the depth function becomes exponentially decaying. For $E_n > V_\infty$ the waves are always travelling towards $z \rightarrow \infty$, hence leaking into the bottom. This is known as "leaky modes".

The mode cut-off occurs when $E_n > \omega^2/c_0^2$, which is higher than the margin of the continuum since the latter is given by $E_n > V_\infty$.

From this short description it is generally concluded that in shallow water the boundary conditions at the bottom might cause an early cutoff, and only few modes propagate. In deep water at very low frequencies only few modes are trapped in the "potential pot". In both cases ray theory cannot describe accurately the acoustic propagation.

APPENDIX E

LISTINGS OF THE COMPUTER PROGRAMS USED FOR THE SIGNAL PROCESSING

E.1 MAIN PROGRAM FOR OFF-LINE PROCESSING

The purpose of the off-line processing is described in Chapter 3. The following is a listing of its main program.

```

0001 FTN4,L
0002 $EMA(RBE,5)
0003 PROGRAM LUNDE(3,99),**** SPG BEAMFORMING 3/3/81 *****
0004 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0005 C
0006 C PROGRAM FOR CONVENTIONAL AND MAXIMUM LIKELIHOOD BEAMFORMING C
0007 C COMBINED WITH DIFFERENT METHODS FOR OWN VESSEL NOISE REDUCTION C
0008 C
0009 C DEVELOPED BY E. B. LUNDE (SPG) C
0010 C MARCH 1981 C
0011 C C
0012 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0013 C
0014 C ARRAYS ARE DEFINED SO THAT THERE CAN BE MAXIMUM
0015 C 19 ARRAY HYDROPHONES + 1 REFERENCE HYDROPHONE,
0016 C 181 BEAMS, 10 FREQUENCIES, AND 10 PROCESSING MODES
0017 C
0018 C PARAMETER VALUES :
0019 C
0020 C NH - NUMBER OF HYDROPHONES
0021 C NB - NUMBER OF BEAMS
0022 C NF - NUMBER OF FREQUENCY CELLS
0023 C F - ARRAY WITH FREQUENCY VALUES
0024 C C - SOUND SPEED (m/sec)
0025 C A - HYDROPHONE SEPARATION (m)
0026 C NA - NUMBER OF AVERAGE IN THE RAW MATRIX
0027 C NT - NUMBER OF TOTAL AVERAGES PER TAPE
0028 C
0029 C NH,NB,NF,F,A,C PARAMETERS WILL BE SET BY THE
0030 C ACQUISITION PROGRAM AND WILL BE IN THE MAGNETIC
0031 C TAPE HEADER BLOCK.
0032 C
0033 C PARAMETERS NA , NT WILL BE TYPED BY THE OPERATOR .
0034 C
0035 C V. DUARTE (COM) 05/03/81/
0036 C
0037 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0038 C
0039 C DIMENSION R(400,2),E(400,2),EV(20,4)
0040 C DIMENSION BGR(182),PR(20,6)
0041 C DIMENSION B(20,2),BE(20,2),BSQ(20)
0042 C DIMENSION NN(5),IDCB(144)
0043 C DIMENSION TAB(181,2)
0044 C DIMENSION P(182,10),PSP(6)
0045 C DIMENSION F(10),KMAX(20),H(20,2,2)
0046 C DIMENSION IB(815),FIB(12),RB(183)
0047 C DIMENSION RSP(36,2),BSP(11,11)
0048 C EQUIVALENCE (P(1,1),E(1,1),IDCB(1)),(P(1,6),R(1,1))
0049 C EQUIVALENCE (IB(1),RB(1),H(1,1,1)),(IB(19),FIB(1))
0050 C EQUIVALENCE (IB(161),B(1,1)),(IB(241),BE(1,1))
0051 C EQUIVALENCE (IB(321),BSQ(1)),(IB(361),EV(1,1))
0052 C EQUIVALENCE (IB(521),BGR(1)),(BSP(1,1),PR(1,1))
0053 C
0054 C COMMON/RBE/LREC(15),TREC(400),RF(2,4000),
0055 C $BEW(4000,2),BSQW(4000)

```

```
C
0057 DATA NP/10/
0058 C
0059 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0060 C
0061 C
0062 C
0063 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0064 C
    10 CONTINUE
    CALL EXEC(3,407B)
    CALL EXEC(1,7,IB,527)
    WRITE(1,1000) (IB(I),I=10,13)
    1000 FORMAT(" TAPE HEADER : ",4A2," OK ? (YE/NO) _")
    READ(1,1001) IP
    1001 FORMAT(2A2)
    IF(IP.EQ.2HYES) GO TO 30
    WRITE(1,1002)
    1002 FORMAT(/," INSERT NEW TAPE AND TYPE "GO" TO PROCEED _")
    20 CONTINUE
    READ(1,1001) IP
    IF(IP.EQ.2HGOD) GO TO 10
    GO TO 20
    30 CONTINUE
    WRITE(1,1035)
    1035 FORMAT(/," MAG TAPE NUMBER ? _")
    READ(1,*) MGTP
    NH=IB(16)-1
    NB=IB(17)
    NF=IB(18)
    A=FIB(NF+1)
    C=FIB(NF+2)
    WRITE(1,1003) NH,NB,NF,A,C
    1003 FORMAT(/," NUMBER OF HYDROPHONES = ",I2,/,"
$"   "NUMBER OF BEAMS = ",I3,/,"
$"   "NUMBER OF FREQUENCIES = ",I2,/,"
$"   "HYDROPHONE SEPARATION = ",F6.4,/,"
$"   "SPEED OF SOUND (m/s) = ",F12.4,/)"
DO 40 I=1,NF
    F(I)=FIB(I)
    WRITE(1,1004) I,F(I)
    1004 FORMAT(2X,I2,4X,F8.4," Hz ")
    40 CONTINUE
    WRITE(1,1005)
    1005 FORMAT(/," OK ? (YE/NO) _")
    READ(1,1001) IP
    IF(IP.NE.2HYES) STOP 001
    4141 CONTINUE
    WRITE(1,1006)
    1006 FORMAT(/," ENTER STARTING BLOCK NUMBER ON M.T. _")
    READ(1,*) IC
    IF(IC.GT.11400) GO TO 4141
    1106 CONTINUE
    WRITE(1,1007)
    1007 FORMAT(/," NUMBER OF AVERAGES ? _")
    READ(1,*) NA
    IF(NA.GE.20) GO TO 50
    WRITE(1,1107)
    1107 FORMAT(/," INPUT ERROR ",/, " 20<= NA <=600 ")
    GO TO 1106
    50 CONTINUE
    WRITE(1,1008)
    1008 FORMAT(/," NUMBER OF AVERAGES PER TAPE ? _")
    READ(1,*) NT
    IF(NT.LE.30.AND.NT.GT.0) GO TO 1108
    WRITE(1,1118)
    1118 FORMAT(/," INPUT ERROR ",/, " 0< NA*NT <= 600 ")
    GO TO 1106
    1108 CONTINUE
```

```

0125 C
0126 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0127 C
0128 C      WRITE ON FILE " >LUNDE " THE FOLLOWING INFO. :
0129 C
0130 C      NT - NUMBER OF AVARAGES PER TAPE
0131 C      NF -      "      OF FREQUENCIES
0132 C      NP -      "      OF PROCESSINGS
0133 C      F1 - FREQUENCY # 1
0134 C      :      :      :
0135 C      Fn -      "      " n      FOR n=NF
0136 C
0137 C      THIS FILE WILL BE READ BY THE PLOTTING PROGRAM
0138 C
0139 C
0140 C      CALL OPEN(IDC.B,IERR,6H>LUNDE)
0141 C      IF(IERR.LT.0) STOP 100
0142 C      CALL CODE
0143 C      WRITE(IB,1109) NT,NF,NP
0144 C      1109 FORMAT(2X,3I4)
0145 C      CALL WRITF(IDC.B,IERR,IB,8)
0146 C      IF(IERR.LT.0) STOP 101
0147 C      CALL CODE
0148 C      WRITE(IB,1119) (F(I),I=1,10)
0149 C      1119 FORMAT(10(2X,F6.2))
0150 C      CALL WRITF(IDC.B,IERR,IB,NF*4)
0151 C      IF(IERR.LT.0) STOP 102
0152 C      CALL CLOSE(IDC.B)
0153 C
0154 C
0155 C      END OF INPUT PARAMETER PHASE [VD].
0156 C
0157 C
0158 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0159 C
0160 C      SOME USEFUL CONSTANTS
0161 C
0162 C      NFE=NF+1
0163 C      NBE=NB+1
0164 C      NBR=NB-1
0165 C      NBH=NB/2+1
0166 C      NHE=NH+1
0167 C      NHD=NH*2
0168 C      NHED=NHE*2
0169 C      NHSG=NH*NH
0170 C      NHESG=NHE*NHE
0171 C      NSP=NH/2
0172 C      ALN10=ALOG(10.)
0173 C
0174 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0175 C
0176 C      CALCULATE TABLE USED FOR STEERING VECTOR GENERATION
0177 C
0178 C      PI=4.*ATAN(1.)
0179 C      ARG=2.*PI/(1.*NBR)
0180 C      SNH=1./SQRT(1.*NH)
0181 C      DO 81 JB=1,NB
0182 C      JBM=JB-1
0183 C      TAB(JB,1)=COS(ARG*JBM)*SNH
0184 C      TAB(JB,2)=SIN(ARG*JBM)*SNH
0185 C      81 CONTINUE
0186 C
0187 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0188 C      FIND THE MAG TAPE START POINT
0189 C
0190 C
0191 C      IF(IC.LE.1) GO TO 83
0192 C      82 CONTINUE
0193 C      CALL EXEC(1,7,IB,815)
0194 C      IF(IB(1).EQ.IC) GO TO 83
0195 C      GO TO 82
0196 C      83 CONTINUE
0197 C

```



```

0198 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0199 C
0200 C      START OF MAIN LOOP
0201 C
0202 C      DO 191 JT=1,NT
0203 C
0204 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0205 C
0206 C      CLEAR THE CORRELATION MATRICES IN EMA
0207 C
0208 C      CALL WSMY(0.,RF(1,1),1,RF(1,1),1,8000)
0209 C
0210 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0211 CCC
0212 CCC      READ RECORDS FROM TAPE
0213 CCC
0214 C      DO 101 JA=1,NA
0215 C      CALL LEMA1(7,IERR)
0216 C      IF(IERR.LT.0) STOP 555
0217 CCC
0218 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0219 C
0220 C      SUBTRACT 1/48 OF X(NH) FROM X(NHE), THIS BECAUSE OF
0221 C      THE METHOD USED IN THE BEAMFORMER TO DELAY X(NHE)
0222 C
0223 C      G=-1./48.
0224 C      DO 22 KA=1,2
0225 C      LM=NHD+KA
0226 C      LH=LM-2
0227 C      CALL WPIV(G,TREC(LH),NHED,TREC(LM),NHED,TREC(LM),NHED,NF)
0228 C      22 CONTINUE
0229 C
0230 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0231 C
0232 C      CALCULATE THE OUTER PRODUCT OF THE VECTOR X
0233 C      AND ADD THE RESULTING MATRIX TO MATRIX RF
0234 C
0235 C      DO 23 JF=1,NF
0236 C      LF=(JF-1)*NHED+1
0237 C      MF=(JF-1)*NHESQ+1
0238 C      CALL WVMOV(TREC(LF),1,R(1,1),1,NHED)
0239 C      DO 24 JH=1,NHE
0240 C      LH=(JH-1)*2
0241 C      LFH=LF+LH
0242 C      MFH=MF+(JH-1)*NHE
0243 C      YR=R(LH+1,1)
0244 C      YI=R(LH+2,1)
0245 C      CALL WPIV(YR,TREC(LF),1,RF(1,MFH),1,RF(1,MFH),1,NHED)
0246 C      CALL WPIV(YI,TREC(LF+1),2,RF(1,MFH),2,RF(1,MFH),2,NHE)
0247 C      CALL WPIV(-YI,TREC(LF),2,RF(2,MFH),2,RF(2,MFH),2,NHE)
0248 C      24 CONTINUE
0249 C      23 CONTINUE
0250 C      101 CONTINUE
0251 C
0252 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0253 C
0254 C      START OF FREQUENCY LOOP
0255 C
0256 C      DO 21 JF=1,NF
0257 C
0258 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0259 C
0260 C      READ THE CORR MATRIX AT THE ACTUAL FREQUENCY FROM EMA
0261 C      TO THE MATRIX R, NORMALIZE AND MOVE IT TO MATRIX E
0262 C
0263 C      LF=(JF-1)*NHESQ+1
0264 C      DO 115 KA=1,2
0265 C      CALL WVMOV(RF(KA,LF),2,R(1,KA),1,NHESQ)
0266 C      115 CONTINUE
0267 C      CALL VSUM(TR,R(1,1),NHE+1,NH)

```

```

0268      POW=TR/(1.*NH)
0269      POWI=1./POW
0270      DO 111 JH=1,NHE
0271      LH=(JH-1)*NHE+1
0272      MH=(JH-1)*NH+1
0273      CALL VSMY(POWI,R(LH,1),1,E(MH,1),1,NH)
0274      CALL VSMY(-POWI,R(LH,2),1,E(MH,2),1,NH)
0275      113 CONTINUE
0276      111 CONTINUE
0277      POW=POW/(1.*NA)
0278      QS=R(NHESQ,1)*POWI
0279      CALL VSMY(100.,E(1,1),NHE,PR(1,2),1,NH)
0280      PR(NHE,2)=QS*100.
0281      PR(NHE,1)=POW
0282      C
0283      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0284      C
0285      C      PRELIMINARY SPLIT BEAM PROCESSING
0286      C
0287      NN(1)=-NH+1
0288      NN(2)=-NH+NSP*NH+NSP+1
0289      NN(3)=-NH+NSP*NH-NSP-1
0290      NN(4)=-NH+NSP*NH+1
0291      DO 701 JSP=1,4
0292      LINC=NHE
0293      IF (JSP.EQ.3) LINC=-NHE
0294      LSP=(JSP-1)*NSP
0295      DO 702 JH=1,NSP
0296      NN(JSP)=NN(JSP)+NH
0297      NEL=NSP+1-JH
0298      IF (JSP.EQ.3) NEL=JH-1
0299      DO 703 KA=1,2
0300      CALL VSUM(RSP(JH+LSP,KA),E(NN(JSP),KA),LINC,NEL)
0301      703 CONTINUE
0302      702 CONTINUE
0303      701 CONTINUE
0304      C      WRITE(6,822) ((RSP(JSP,KA),KA=1,2),JSP=1,36)
0305      822 FORMAT(2F10.3)
0306      C
0307      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0308      C
0309      C      USE A SUBROUTINE TO FIND EIGENVALUES AND EIGENVECTORS OF R
0310      C      THE EIGENVALUES ARE STORED IN DECREASING ORDER IN VECTOR EV
0311      C      AND THE RESPECTIVE EIGENVECTORS IN MATRIX E
0312      C
0313      NH1=NH
0314      CALL EIGEN(E(1,1),E(1,2),R(1,1),R(1,2),NHSQ,EV(1,1),NH1)
0315      CALL VSMY(10.,EV(1,1),1,PR(1,1),1,NH)
0316      C
0317      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0318      C
0319      C      INVERT THE EIGENVALUES
0320      C
0321      CALL VSDV(1.,EV(1,1),1,EV(1,2),1,NH)
0322      C
0323      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0324      C
0325      C      START OF PRELIMINARY BEAM LOOP
0326      C
0327      DO 41 JB=1,NBE
0328      IF (JB.EQ.NBE) GO TO 223
0329      C
0330      C
0331      C      CREATE STEERING VECTOR FROM TABLE
0332      C
0333      NBH=NB/2+1
0334      K=NBH-JB
0335      IF (K.LT.0) K=K+NBR
0336      L=0
0337      DO 33 JH=1,NH
0338      DO 32 KA=1,2
0339      B(JH,KA)=TAB(L+1,KA)

```

```

0340      32 CONTINUE
0341      L=L+K
0342      IF (L. GE. NBR) L=L-NBR
0343      33 CONTINUE
0344      C
0345      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0346      C
0347      C      CALCULATE BEAM DIRECTION OR RELATIVE WAVENUMBER
0348      C
0349      AL=A*(JF)/C
0350      JJ=1
0351      ARG=(NBH-JB)/(AL*NBR)
0352      225 CONTINUE
0353      AARG=ABS(ARG)
0354      IF (AARG. LE. 1.) GO TO 222
0355      BGR(JB)=-10.*AARG
0356      GO TO 223
0357      222 CONTINUE
0358      ARH=SQRT(1.-ARG*ARG)
0359      BGR(JB)=ATAN2(ARH, ARG)*180./PI
0360      223 CONTINUE
0361      C      IF (JJ. EQ. 2) GO TO 224
0362      C      JJ=2
0363      C      IF (JB. LE. NBH) ARG=ARG-1./AL
0364      C      IF (JB. GT. NBH) ARG=ARG+1./AL
0365      C      GO TO 225
0366      C      224 CONTINUE
0367      C
0368      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0369      C
0370      C      CALCULATE THE VECTORS B'E, B'G, Q'E, AND Q'G, AND STORE IN EMA
0371      C
0372      IF (JB. NE. NBE) GO TO 47
0373      DO 48 KA=1,2
0374      CALL VMOV(E(NHSQ+1, KA), 1, B(1, KA), 1, NH)
0375      C      NSB(NBE, KA)=0
0376      48 CONTINUE
0377      47 CONTINUE
0378      LB=(JB-1)*NHE+1
0379      DO 42 JH=1, NHE
0380      LH=(JH-1)*NH+1
0381      DO 43 KA=1,2
0382      DO 44 KB=1,2
0383      CALL VDOT(H(JH, KA, KB), B(1, KA), 1, E(LH, KB), 1, NH)
0384      44 CONTINUE
0385      43 CONTINUE
0386      42 CONTINUE
0387      CALL VADD(H(1, 1, 1), 1, H(1, 2, 2), 1, BE(1, 1), 1, NHE)
0388      CALL VSUB(H(1, 2, 1), 1, H(1, 1, 2), 1, BE(1, 2), 1, NHE)
0389      DO 45 KA=1,2
0390      CALL VMOV(BE(1, KA), 1, BEW(LB, KA), 1, NHE)
0391      45 CONTINUE
0392      C
0393      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0394      C
0395      C      CALCULATE THE SQUARE AMPLITUDE OF EACH ELEMENT OF MATRIX BE
0396      C      AND STORE IN EMA
0397      C
0398      DO 46 KA=1,2
0399      CALL VMPY(BE(1, KA), 1, BE(1, KA), 1, H(1, KA, 1), 1, NHE)
0400      46 CONTINUE
0401      CALL VADD(H(1, 1, 1), 1, H(1, 2, 1), 1, BSQ(1), 1, NHE)
0402      CALL VMOV(BSQ(1), 1, BSQW(LB), 1, NHE)
0403      P(JB, 7)=100.*BSQ(1)
0404      P(JB, 8)=100.*BSQ(2)
0405      P(JB, 9)=100.*BSQ(NHE)
0406      IF (JB. LT. NBE) GO TO 715
0407      QNRM=SQRT(BSQ(NHE))
0408      QNRMI=1./QNRM
0409      CALL VSMY(QNRMI, P(1, 9), 1, P(1, 9), 1, NB)
0410      CALL VSMY(QNRMI*100., BSQ(1), 1, PR(1, 5), 1, NHE)
0411      715 CONTINUE
0412      C      WRITE(6, 819) JB, WA, WB, (BSQ(JH), JH=1, NHE), JB

```

```

0413 C B19 FORMAT(X, I3, 2F8. 1, 2X, 20F4. 3, 2X, I6)
0414 C
0415 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0416 C
0417 C      CONTINUE SPLIT BEAM PROCESSING
0418 C
0419 C      DO 711 JSP=1, 3
0420 C      NEL=NSP-1
0421 C      IF (JSP.EQ. 3) NEL=NSP*2-1
0422 C      LSP=(JSP-1)*NSP+1
0423 C      DO 712 KA=1, 2
0424 C      DO 713 KB=1, 2
0425 C      CALL VDOT(H(JSP, KA, KB), B(2, KA), 1, RSP(LSP+1, KB), 1, NEL)
0426 C      713 CONTINUE
0427 C      712 CONTINUE
0428 C      PSP(JSP)=RSP(LSP, 1)/(1. *NH)+2. *(H(JSP, 1, 1)-H(JSP, 2, 2))*SNH
0429 C      711 CONTINUE
0430 C      PSP(3)=PSP(3)/2.
0431 C      PSP(4)=(H(3, 2, 1)+H(3, 1, 2))*SNH
0432 C      PSP(5)=PSP(3)/SQRT(PSP(1)*PSP(2))
0433 C      PSP(6)=PSP(1)+PSP(2)+2. *PSP(3)
0434 C      P(JB, 6)=PSP(3)*100.
0435 C      P(JB, 10)=1.
0436 C      IF (BGR(JB).LT. 0.) GO TO 718
0437 C      DEL=ATAN2(PSP(4), PSP(3))/(2. *PI*AL*NSP)
0438 C      DCOS=COS(BGR(JB)*PI/180.)
0439 C      RCOS=DCOS-DEL
0440 C      IF (ABS(RCOS).GT. 1.) GO TO 718
0441 C      RSIN=SQRT(1.-RCOS*RCOS)
0442 C      P(JB, 10)=ATAN2(RSIN, RCOS)*180. /PI
0443 C      718 CONTINUE
0444 C      IF (ISSW(3).GE. 0) GO TO 719
0445 C      WRITE(6, B17) JB, BGR(JB), (PSP(JM), JM=1, 6), JB
0446 C      B17 FORMAT(X, I3, 4X, F10. 2, 4X, 6F10. 4, 4X, I6)
0447 C      719 CONTINUE
0448 C
0449 C      41 CONTINUE
0450 C
0451 C      END OF PRELIMINARY BEAM LOOP
0452 C
0453 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0454 C      WRITE(6, B18) JA, JF, POW, QNRM, QS
0455 C      B18 FORMAT(2I10, 3F10. 3)
0456 C      IF (ISSW(3).LT. 0) WRITE(6, 535)
0457 C
0458 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0459 C
0460 C      FIND THE BEAM-INDEX ASSOCIATED WITH EACH EIGENVALUE
0461 C
0462 C      DO 51 JH=1, NHE
0463 C      CALL WMAX(KMAX(JH), BSQW(JH), NHE, NB)
0464 C      PR(JH, 3)=KMAX(JH)
0465 C      PR(JH, 4)=BGR(KMAX(JH))
0466 C      51 CONTINUE
0467 C
0468 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0469 C
0470 C      PRINT THE EIGENVALUES AND THE ASSOCIATED BEAM-INDECES
0471 C
0472 C      WRITE(6, B10) (JH, JH=1, NH)
0473 C      WRITE(1, B10) (JH, JH=1, NH)
0474 C      WRITE(6, B10) (KMAX(JH), JH=1, NH)
0475 C      WRITE(1, B10) (KMAX(JH), JH=1, NH)
0476 C      WRITE(6, B10) (PR(JH, 1), JH=1, NH)
0477 C      WRITE(1, B10) (PR(JH, 1), JH=1, NH)
0478 C      B10 FORMAT(X, 19I4)
0479 C      B11 FORMAT(X, F5. 3, 18F4. 3)
0480 C
0481 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0482 C
0483 C      FIND REF BEAM DIRECTION
0484 C

```

```

0485      BDEL=-1.
0486      DO 56 JB=1,NB
0487      IF (BGR(JB).LE.O.) GO TO 56
0488      ADEL=P(JB,10)-BGR(JB)
0489      IF (ADEL.LE.O..AND.BDEL.GT.O.) GO TO 57
0490      BDEL=ADEL
0491      56 CONTINUE
0492      57 CONTINUE
0493      KREF=JB
0494      IF (BDEL.LT.-ADEL) KREF=JB-1
0495      C
0496      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0497      C
0498      C      FIND THE EIGENVALUE(S) TO BE MODIFIED
0499      C      AND MODIFY THEM, THEN INVERT THEM
0500      C
0501      CALL VMOV(EV(1,1),1,EV(1,3),1,NH)
0502      JJ=11
0503      DO 444 JH=1,NH
0504      IF (IABS(KMAX(JH)-KREF).GT.3) GO TO 444
0505      EV(JH,3)=EV(NH,3)
0506      PR(JJ,6)=JH
0507      JJ=JJ+1
0508      444 CONTINUE
0509      CALL VSDV(1.,EV(1,3),1,EV(1,4),1,NH)
0510      C
0511      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0512      C
0513      C      CALCULATE CONVENTIONAL AND MAXIMUM LIKELIHOOD BEAMFORMING
0514      C
0515      DO 61 JB=1,NBE
0516      LB=(JB-1)*NHE+1
0517      CALL WVMOV(BSQW(LB),1,BSQ(1),1,NHE)
0518      DO 62 JM=1,4
0519      CALL VDOT(P(JB,JM),EV(1,JM),1,BSQ(1),1,NH)
0520      62 CONTINUE
0521      DO 66 JH=1,NH
0522      KMAX(JH)=100.*BSQ(JH)
0523      IF (JB.EQ.NBE) KMAX(JH)=100.*BSQ(JH)/GNRM
0524      66 CONTINUE
0525      KMAX(NHE)=100.*BSQ(NHE)/GNRM
0526      IF (JB.EQ.NBE) KMAX(NHE)=100.*BSQ(NHE)/(GNRM*GNRM)
0527      IF (ISSW(0).GE.0) GO TO 69
0528      WRITE(6,815) JB,BGR(JB),(KMAX(JH),JH=1,NHE),JB
0529      815 FORMAT(X,13,4X,F10.2,4X,2014,4X,I6)
0530      69 CONTINUE
0531      61 CONTINUE
0532      CALL VMIN(MIN,P(1,1),1,NB)
0533      CALL VSDV(1.,P(1,2),1,P(1,2),1,NB)
0534      CALL VSDV(1.,P(1,4),1,P(1,4),1,NB)
0535      QR=QS-P(NBE,2)
0536      C      WRITE(6,845) QS,P(NBE,2),QR
0537      C 845 FORMAT(3F10.3)
0538      IF (ISSW(0).LT.0) WRITE(6,535)
0539      C
0540      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0541      C
0542      C      MAIN PRINT OUTPUT
0543      C
0544      DO 536 KA=1,2
0545      CALL VSMY(100.,P(1,KA),1,P(1,KA),1,NB)
0546      536 CONTINUE
0547      PR(1,6)=IC
0548      PR(2,6)=NA
0549      PR(3,6)=JT
0550      PR(4,6)=JF
0551      PR(5,6)=F(JF)
0552      PR(6,6)=POW
0553      PR(7,6)=GNRM*100./(1.*NH)
0554      PR(8,6)=QS*100.
0555      PR(9,6)=QR*100.
0556      PR(10,6)=BGR(KREF)
0557      PR(20,6)=MGTP

```



```

0558      PR(19,6)=C
0559      PR(18,6)=F(JF)*10.
0560      PR(17,6)=A
0561      DO 876 JH=JJ,16
0562      PR(JH,6)=0.
0563 876 CONTINUE
0564      WRITE(6,538)
0565 538 FORMAT(12X,3(4X," # BEARING CON MLM SPL BE1 BE2 BQ"))
0566      DO 531 I=1,3
0567      M=(I-1)*2+1
0568      DO 533 J=1,20
0569      IJ=(I-1)*20+J
0570      DO 534 K=1,3
0571      NN(K)=(K-1)*60+IJ
0572 534 CONTINUE
0573
0574      WRITE(6,9) J,PR(J,M),PR(J,M+1),(NN(K),BGR(NN(K)),P(NN(K),10),
0575      (P(NN(K),L),L=1,2),(P(NN(K),L),L=6,9),K=1,3)
0576      9 FORMAT(3I4,3(4X,9I4))
0577 533 CONTINUE
0578 531 CONTINUE
0579      DO 537 KA=1,2
0580      CALL VSMY(.01,P(1,KA),1,P(1,KA),1,NB)
0581 537 CONTINUE
0582      DO 676 JB=1,NB
0583      PH=2.5*ALN10/(1+.5*ABS(P(JB,10)-BGR(JB)))
0584      P(JB,9)=EXP(PH*P(JB,6)/250.)
0585      P(JB,10)=EXP(PH)
0586 676 CONTINUE
0587      WRITE(6,535)
0588 535 FORMAT(1H1)
0589 C
0590 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0591 C
0592 C SELECT REFERENCE BEAM DIRECTION AND DO REFERENCE
0593 C BEAM NOISE SUPPRESSION .
0594 C
0595 C NREF - REFERENCE BEAM NUMBER
0596 C
0597      NREF=KREF
0598      LREF=(NREF-1)*NHE+1
0599      DO 96 KA=1,2
0600      CALL WVMOV(BEW(LREF,KA),1,BE(1,KA),1,NH)
0601 96 CONTINUE
0602      DO 97 JB=1,NB
0603      LB=(JB-1)*NHE+1
0604      DO 98 KA=1,2
0605      CALL WVMOV(BEW(LB,KA),1,B(1,KA),1,NH)
0606      DO 99 KB=1,2
0607      CALL VMPY(B(1,KA),1,BE(1,KB),1,H(1,KA,KB),1,NH)
0608 99 CONTINUE
0609 98 CONTINUE
0610      CALL VADD(H(1,1,1),1,H(1,2,2),1,H(1,1,1),1,NH)
0611      CALL VSUB(H(1,1,2),1,H(1,2,1),1,H(1,2,1),1,NH)
0612      DO 294 KM=1,2
0613      JM=KM+4
0614      DO 293 KA=1,2
0615      CALL VDOT(H(KA,1,2),EV(1,KM),1,H(1,KA,1),1,NH)
0616 293 CONTINUE
0617      CALL VDOT(P(JB,JM),H(1,1,2),1,H(1,1,2),1,2)
0618 294 CONTINUE
0619 97 CONTINUE
0620      SC=1./P(NREF,1)
0621      CALL VSMY(SC,P(1,5),1,P(1,5),1,NB)
0622      CALL VSMY(P(NREF,2),P(1,6),1,P(1,6),1,NB)
0623      CALL VMPY(P(1,6),1,P(1,2),1,P(1,6),1,NB)
0624      CALL VMPY(P(1,6),1,P(1,2),1,P(1,6),1,NB)
0625      DO 292 KM=1,2
0626      JM=KM+4
0627      CALL VSUB(P(1,KM),1,P(1,JM),1,P(1,JM),1,NB)
0628 292 CONTINUE
0629 C

```

```

0630 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0631 C
0632 C      DO REFERENCE HYDROPHONE NOISE SUPPRESSION
0633 C
0634      CALL VMOV(1., 0, EV(1, 1), 1, NH)
0635      LREF=NB*NHE+1
0636      DO 601 KA=1, 2
0637      CALL WVMOV(BEW(LREF, KA), 1, BE(1, KA), 1, NH)
0638 601 CONTINUE
0639      DO 602 JB=1, NB
0640      LB=(JB-1)*NHE+1
0641      DO 603 KA=1, 2
0642      CALL WVMOV(BEW(LB, KA), 1, B(1, KA), 1, NH)
0643      DO 604 KB=1, 2
0644      CALL VMPY(B(1, KA), 1, BE(1, KB), 1, H(1, KA, KB), 1, NH)
0645 604 CONTINUE
0646 603 CONTINUE
0647      CALL VADD(H(1, 1, 1), 1, H(1, 2, 2), 1, H(1, 1, 1), 1, NH)
0648      CALL VSUB(H(1, 1, 2), 1, H(1, 2, 1), 1, H(1, 2, 1), 1, NH)
0649      DO 605 KM=1, 2
0650      JM=KM+6
0651      DO 606 KA=1, 2
0652      CALL VDOT(H(KA, 1, 2), EV(1, KM), 1, H(1, KA, 1), 1, NH)
0653 606 CONTINUE
0654      CALL VDOT(P(JB, JM), H(1, 1, 2), 1, H(1, 1, 2), 1, 2)
0655 605 CONTINUE
0656 602 CONTINUE
0657      SC=1./QS
0658      CALL VSMY(SC, P(1, 7), 1, P(1, 7), 1, NB)
0659      SC=-1./GR
0660      CALL VSMY(SC, P(1, 8), 1, P(1, 8), 1, NB)
0661      CALL VSDV(1., P(1, 2), 1, P(1, 2), 1, NB)
0662      DO 607 KM=1, 2
0663      JM=KM+6
0664      CALL VSUB(P(1, KM), 1, P(1, JM), 1, P(1, JM), 1, NB)
0665 607 CONTINUE
0666      CALL VSDV(1., P(1, 2), 1, P(1, 2), 1, NB)
0667      CALL VSDV(1., P(1, 8), 1, P(1, 8), 1, NB)
0668      IF (ISSW(1).GE.0) GO TO 609
0669      DO 281 JB=1, NB
0670      WRITE(6, 816) JB, BGR(JB), (P(JB, JM), JM=1, 10), JB
0671 816 FORMAT(X, I3, 4X, F10.2, 4X, 10F10.4, 4X, I6)
0672 281 CONTINUE
0673      WRITE(6, 535)
0674 609 CONTINUE
0675 C
0676 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0677 C
0678 C      MOVE P(JB, JM), JB=1, NB AND JM=1, 2 TO DISC
0679 C
0680      JT1=JT
0681      IF(JT.GT.16) JT1=JT-16
0682      LD=(JF-1)*2
0683      ISEC=(JT1-1)*6
0684      ITRK1=LD*NP
0685      DO 400 JD=1, NP
0686      ITRK=ITRK1+(JD-1)*2
0687      IF(JT.GT.16) ITRK=ITRK+1
0688 C      DO 500 ID=1, NB
0689 C      RB(ID)=P(ID, JD)
0690 C 500 CONTINUE
0691 C      WRITE(6, 599) (P(JB, JD), JB=1, NBE)
0692      CALL INTER(P(1, JD), RB(1), F(JF), NB)
0693      CALL VMAX(NMAX, RB(1), 1, NB)
0694      RB(182)=RB(NMAX)
0695      RB(183)=POW
0696 C      NBC=NB+2
0697 C      WRITE(6, 599) (RB(JB), JB=1, NBC)
0698 C 599 FORMAT(10F8.3)
0699      CALL EXEC(2, 14, IB, 384, ITRK, ISEC)
0700 400 CONTINUE

```

```
0701 C  
0702      21 CONTINUE  
0703 C  
0704 C      END OF FREQUENCY LOOP  
0705 C  
0706 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
0707 C  
0708 C  
0709 C  
0710 C      END OF MAIN LOOP  
0711 C  
0712      191 CONTINUE  
0713 C  
0714 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
0715 C  
0716 C      ALL DATA ARE PROCESSED AND THE RESULTS ARE ON DISC  
0717 C  
0718 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
0719 C  
0720 C      END OF MAIN PROGRAM  
0721 C  
0722       IMERD=LREC(1)  
0723       WRITE(1,9191) IMERD  
0724   9191 FORMAT(/," LAST BLOCK : ",I6)  
0725 C  
0726 C  
0727       END
```

FTN4 COMPILER: HP92060-16092 REV. 2001 (791101)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 10953 COMMON = 00000

E.2 SUBROUTINE EIGEN

The EIGEN subroutine calculates the eigenvalues and eigenvectors of a Hermitian matrix. A listing of the subroutine follows.

```

0001  FTN4,L
0002      SUBROUTINE EIGEN(ARE, AIM, ERE, EIM, ISIZE, EIGV, N)
0003      DIMENSION ARE(ISIZE), AIM(ISIZE), ERE(ISIZE), EIM(ISIZE)
0004      DIMENSION EIGV(20), H(20, 4)
0005      INTEGER P, Q, NN(20)
0006  C
0007  C      DATA FOR BREAKING OFF
0008  C
0009      NNN=N
0010  C
0011      IDW=N
0012  C
0013      EPS=. 1E-9
0014  C
0015  C      D  UNITY MATRIX ERE, EIM, FOR EIGENVECTORS
0016  C
0017      DO 1 I=1, N
0018      DO 1 J=1, N
0019      IJ=(I-1)*N+J
0020      JI=(J-1)*N+I
0021      ERE(IJ)=0.
0022      EIM(IJ)=0.
0023      IF (I. GT. J) ARE(IJ)=ARE(JI)*ARE(JI)+AIM(JI)*AIM(JI)
0024 1  CONTINUE
0025      DO 2 I=1, N
0026      K9=(I-1)*N+I
0027      ERE(K9)=1.
0028 2  CONTINUE
0029  C
0030  C      BEGIN OF ITERATION
0031  C
0032      DO 100 IT=1, 1000
0033  C
0034  C      COMPUTATION OF SQUARE SUM OF NONDIAGONAL ELEMENTS
0035  C      AND DETERMINATION OF MAX NONDIAGONAL ELEMENT
0036  C
0037  C      SW=0.
0038      H(1, 1)=0.
0039      DO 101 I=2, N
0040      NEL=I-1
0041      IJ=(I-1)*N+1
0042      CALL VMAX(NN(I), ARE(IJ), 1, NEL)
0043      H(I, 1)=ARE(IJ-1+NN(I))
0044 101 CONTINUE
0045      CALL VMAX(Q, H(1, 1), 1, N)
0046      P=NN(Q)
0047      SW=SQRT(H(Q, 1))
0048      NNN=IT
0049      IF(SW. LT. EPS) GOTO 899
0050  C
0051  C      PIVOT-ELEMENT FOUND COMPUTATION ROTATION PARAMETERS
0052  C
0053      NP=N*P
0054      NQ=N*Q
0055      NPP=NP-N+P
0056      NPQ=NP-N+Q
0057      NGP=NQ-N+P
0058      NGQ=NQ-N+Q
0059      APP=ARE(NPP)
0060      AGQ=ARE(NGQ)
0061      APQ=ARE(NPQ)
0062      BPQ=AIM(NPQ)
0063      RD=SQRT(APQ*APQ+BPQ*BPQ)
0064      ROD=2. *RD
0065      C2X=(AGQ-APP)/ROD
0066      VZ=-1.

```

```

0067      IF(C2X.GE.0) VZ=1.
0068      CX=C2X+VZ*SQRT(C2X*C2X+1.)
0069      C2X=CX*CX
0070      CC=C2X/(1.+C2X)
0071      C=SQRT(CC)
0072      S=C/CX
0073      SS=S*S
0074      SC=S*C
0075      SR=APG*S/RO
0076      SI=-BPQ*S/RO
0077  C
0078  C      ROTATION
0079  C
0080      NEL=P-1
0081      IP=P
0082      IQ=Q
0083      CALL VSMY(SI, AIM(IQ), N, H(1,1), 1, NEL)
0084      CALL VSMY(-SR, AIM(IQ), N, H(1,2), 1, NEL)
0085      CALL VSMY(SI, AIM(IP), N, H(1,3), 1, NEL)
0086      CALL VSMY(SR, AIM(IP), N, H(1,4), 1, NEL)
0087      CALL VPIV(-SR, ARE(IQ), N, H(1,1), 1, H(1,1), 1, NEL)
0088      CALL VPIV(-SI, ARE(IQ), N, H(1,2), 1, H(1,2), 1, NEL)
0089      CALL VPIV(SR, ARE(IP), N, H(1,3), 1, H(1,3), 1, NEL)
0090      CALL VPIV(-SI, ARE(IP), N, H(1,4), 1, H(1,4), 1, NEL)
0091      CALL VPIV(C, ARE(IP), N, H(1,1), 1, ARE(IP), N, NEL)
0092      CALL VPIV(C, AIM(IP), N, H(1,2), 1, AIM(IP), N, NEL)
0093      CALL VPIV(C, ARE(IQ), N, H(1,3), 1, ARE(IQ), N, NEL)
0094      CALL VPIV(C, AIM(IQ), N, H(1,4), 1, AIM(IQ), N, NEL)
0095      JP=NP-N+1
0096      JQ=NG-N+1
0097      CALL VMPY(ARE(IP), N, ARE(IP), N, ARE(JP), 1, NEL)
0098      CALL VMPY(AIM(IP), N, AIM(IP), N, AIM(JP), 1, NEL)
0099      CALL VMPY(ARE(IQ), N, ARE(IQ), N, ARE(JQ), 1, NEL)
0100      CALL VMPY(AIM(IQ), N, AIM(IQ), N, AIM(JQ), 1, NEL)
0101      CALL VADD(ARE(JP), 1, AIM(JP), 1, ARE(JP), 1, NEL)
0102      CALL VADD(ARE(JQ), 1, AIM(JQ), 1, ARE(JQ), 1, NEL)
0103      NEL=Q-P-1
0104      IP=NPP+1
0105      IQ=NP+Q
0106      CALL VSMY(SI, AIM(IQ), N, H(1,1), 1, NEL)
0107      CALL VSMY(SR, AIM(IQ), N, H(1,2), 1, NEL)
0108      CALL VSMY(-SI, AIM(IP), 1, H(1,3), 1, NEL)
0109      CALL VSMY(-SR, AIM(IP), 1, H(1,4), 1, NEL)
0110      CALL VPIV(-SR, ARE(IQ), N, H(1,1), 1, H(1,1), 1, NEL)
0111      CALL VPIV(SI, ARE(IQ), N, H(1,2), 1, H(1,2), 1, NEL)
0112      CALL VPIV(SR, ARE(IP), 1, H(1,3), 1, H(1,3), 1, NEL)
0113      CALL VPIV(-SI, ARE(IP), 1, H(1,4), 1, H(1,4), 1, NEL)
0114      CALL VPIV(C, ARE(IP), 1, H(1,1), 1, ARE(IP), 1, NEL)
0115      CALL VPIV(C, AIM(IP), 1, H(1,2), 1, AIM(IP), 1, NEL)
0116      CALL VPIV(C, ARE(IQ), N, H(1,3), 1, ARE(IQ), N, NEL)
0117      CALL VPIV(C, AIM(IQ), N, H(1,4), 1, AIM(IQ), N, NEL)
0118      JP=NP+P
0119      JQ=NGP+1
0120      CALL VMPY(ARE(IP), 1, ARE(IP), 1, ARE(JP), N, NEL)
0121      CALL VMPY(AIM(IP), 1, AIM(IP), 1, AIM(JP), N, NEL)
0122      CALL VMPY(ARE(IQ), N, ARE(IQ), N, ARE(JQ), 1, NEL)
0123      CALL VMPY(AIM(IQ), N, AIM(IQ), N, AIM(JQ), 1, NEL)
0124      CALL VADD(ARE(JP), N, AIM(JP), N, ARE(JP), N, NEL)
0125      CALL VADD(ARE(JQ), 1, AIM(JQ), 1, ARE(JQ), 1, NEL)
0126      NEL=N-Q
0127      IP=NPQ+1
0128      IQ=NGQ+1
0129      CALL VSMY(-SI, AIM(IQ), 1, H(1,1), 1, NEL)
0130      CALL VSMY(-SR, AIM(IQ), 1, H(1,2), 1, NEL)
0131      CALL VSMY(-SI, AIM(IP), 1, H(1,3), 1, NEL)
0132      CALL VSMY(SR, AIM(IP), 1, H(1,4), 1, NEL)
0133      CALL VPIV(-SR, ARE(IQ), 1, H(1,1), 1, H(1,1), 1, NEL)
0134      CALL VPIV(SI, ARE(IQ), 1, H(1,2), 1, H(1,2), 1, NEL)
0135      CALL VPIV(SR, ARE(IP), 1, H(1,3), 1, H(1,3), 1, NEL)
0136      CALL VPIV(SI, ARE(IP), 1, H(1,4), 1, H(1,4), 1, NEL)
0137      CALL VPIV(C, ARE(IP), 1, H(1,1), 1, ARE(IP), 1, NEL)
0138      CALL VPIV(C, AIM(IP), 1, H(1,2), 1, AIM(IP), 1, NEL)

```



```

0139      CALL VPIV(C, ARE(IQ), 1, H(1, 3), 1, ARE(IQ), 1, NEL)
0140      CALL VPIV(C, AIM(IQ), 1, H(1, 4), 1, AIM(IQ), 1, NEL)
0141      JP=NG+P
0142      JQ=NG+Q
0143      CALL VMPY(ARE(IP), 1, ARE(IP), 1, ARE(JP), N, NEL)
0144      CALL VMPY(AIM(IP), 1, AIM(IP), 1, AIM(JP), N, NEL)
0145      CALL VMPY(ARE(IQ), 1, ARE(IQ), 1, ARE(JQ), N, NEL)
0146      CALL VMPY(AIM(IQ), 1, AIM(IQ), 1, AIM(JQ), N, NEL)
0147      CALL VADD(ARE(JP), N, AIM(JP), N, ARE(JP), N, NEL)
0148      CALL VADD(ARE(JQ), N, AIM(JQ), N, ARE(JQ), N, NEL)
0149      C
0150      C   E. B. L. 18/02/82
0151      C
0152      ARE(NPP)=APP*CC+AQG*SS-ROD*SC
0153      ARE(NGG)=APP*SS+AQG*CC+ROD*SC
0154      ARE(NPG)=0.
0155      AIM(NPG)=0.
0156      ARE(NGP)=0.
0157      C
0158      C   COMPUTATION OF EIGENVECTORS
0159      C
0160      IP=NP-N+1
0161      IQ=NG-N+1
0162      CALL VSMY(SI, EIM(IQ), 1, H(1, 1), 1, N)
0163      CALL VSMY(-SR, EIM(IQ), 1, H(1, 2), 1, N)
0164      CALL VSMY(SI, EIM(IP), 1, H(1, 3), 1, N)
0165      CALL VSMY(SR, EIM(IP), 1, H(1, 4), 1, N)
0166      CALL VPIV(-SR, ERE(IQ), 1, H(1, 1), 1, H(1, 1), 1, N)
0167      CALL VPIV(-SI, ERE(IQ), 1, H(1, 2), 1, H(1, 2), 1, N)
0168      CALL VPIV(SR, ERE(IP), 1, H(1, 3), 1, H(1, 3), 1, N)
0169      CALL VPIV(-SI, ERE(IP), 1, H(1, 4), 1, H(1, 4), 1, N)
0170      CALL VPIV(C, ERE(IP), 1, H(1, 1), 1, ERE(IP), 1, N)
0171      CALL VPIV(C, EIM(IP), 1, H(1, 2), 1, EIM(IP), 1, N)
0172      CALL VPIV(C, ERE(IQ), 1, H(1, 3), 1, ERE(IQ), 1, N)
0173      CALL VPIV(C, EIM(IQ), 1, H(1, 4), 1, EIM(IQ), 1, N)
0174      1006 FORMAT(2X, 'ARE=', 4(F10.3, 3X), 5X, 'AIM=', 4(F10.3, 3X))
0175      1007 FORMAT(2X, 'ERE=', 4(F10.3, 3X), 5X, 'EIM=', 4(F10.3, 3X))
0176      C
0177      C   ROTATION END
0178      100  CONTINUE
0179      C
0180      C   200 ITERATIONS PERFORMED SET N=-N
0181      C
0182      NNN=-N
0183      C   WRITE(1, 1002) NNN
0184      C1002 FORMAT(10X, 'NNN=-N=', I3)
0185      C
0186      C   SORT OF EIGENVALUES WITH INCREASING ABS VALUES
0187      C
0188      899  LL=1
0189      900  AK=-1. E04
0190          DO 901 I=1, N
0191          K2=(I-1)*N+I
0192      901  AK=AMAX1(AK, ARE(K2))
0193          DO 902 I=1, N
0194          K1=(I-1)*N+I
0195          IF(AK.EQ.ARE(K1)) GOTO 903
0196      902  CONTINUE
0197      C
0198      C   FOUND STORE IN EIGV MARC POSITION
0199      C
0200      903  NN(LL)=I
0201          EIGV(LL)=AK
0202          LL=LL+1
0203          K11=(I-1)*N+I
0204          ARE(K11)=-1. E05
0205          IF(LL-N) 900, 900, 904
0206      C
0207      C   EIGENVALUES ARE SORTED IN EIGV. SORT OF COLUMNS
0208      C   OF ERE, EIM IN THE SAME ORDER AND STORE IN ARE, AIM.
0209      C

```

```

0210 904 DO 905 I=1,N
0211      KO=(I-1)*N+1
0212      K10=(NN(I)-1)*N+1
0213      CALL VMOV(ERE(K10),1,ARE(KO),1,N)
0214      CALL VSMY(-1.,EIM(K10),1,AIM(KO),1,N)
0215 905 CONTINUE
0216      N=NNN
0217      ICL=15414B
0218      IBL=3407B
0219      CALL EXEC(2,1001B,ICL,1)
0220      CALL EXEC(2,1001B,IBL,1)
0221      WRITE(1,1003) N
0222 1003 FORMAT(10X,'OUT OF EIGEN  N=NNN=',I3)
0223      WRITE(1,2004) (EIGV(I),I=1,IOW)
0224 2004 FORMAT(5X,'EIGV= ',F15.5)
0225 C
0226 C RETURN BREAKING OFF
0227 C
0228      RETURN
0229      END

```

FTN4 COMPILER: HP92060-16092 REV. 2001 (791101)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 02935

COMMON = 00000

E.3 ASSEMBLER PROGRAM FOR DATA TRANSFER

The following is a listing of a program for transferring data from magnetic tape into a part of the off-line HP computer memory called "EMA".

```

0001          ASMB,B,L,T
0002 00000          NAM LEMA1,7      **** COM,V. DUARTE REV.00 05/03/81 ****
0003          ENT LEMA1
0004          EXT .EMAP,.ENTR,EXEC
0005          EXT RBE
0006*
0007*****
0008*
0009*          C O M
0010*
0011*          V. DUARTE      MARCH 81      REV 00
0012*
0013*          CALLING SEQUENCE :
0014*
0015*          CALL LEMA1(LU,IERR)
0016*
0017*          LU - MAGNETIC TAPE LOGIC UNIT
0018*          IERR - RETURN IF ERROR
0019*
0020*          PURPOSE :
0021*
0022*          ROUTINE TO TRANSFER DATA FROM MAGNETIC TAPE INTO
0023*          AN "EMA" ARRAY."EMA" ARRAY SHOULD BE DECLARED AS THE
0024*          FIRST OF THE "EMA" AREA.
0025*
0026*****
0027*
0028 00000 000000 LU      NOP          MT LOGIC UNIT
0029 00001 000000 IERR   NOP          ERROR RETURN
0030 00002 000000 LEMA1  NOP          ROUTINE ENTRY POINT
0031 00003 016002X      JSB .ENTR     GET INPUT PARAMETER
0032 00004 000000R      DEF LU
0033 00005 016001X      JSB .EMAP     GET USER MAP ADDRESS
0034 00006 000012R      DEF ++4
0035 00007 000004X      DEF RBE       OF CORRESPONDENT EMA
0036 00010 000032R      DEF TABLE   AREA ARRAY
0037 00011 000037R      DEF INDEX
0038 00012 026030R RTN   JMP ERROR    SEND ERROR MESSAGE TO MAIN
0039 00013 076041R      STB BUF       SAVE ADDRESS
0040 00014 162000R      LDA LU,I      PREPARE CONTROL WORD
0041 00015 042043R      ADA =B100     FOR EXEC CALL AS BINARY READ
0042 00016 072000R      STA LU
0043 00017 016003X      JSB EXEC      READ DATA FROM MT
0044 00020 000025R      DEF ++5
0045 00021 000040R      DEF CODE
0046 00022 000000R      DEF LU
0047 00023 100041R      DEF BUF,I
0048 00024 000042R      DEF NWORD
0049 00025 002400      CLA
0050 00026 172001R ENDI   STA IERR,I
0051 00027 126002R      JMP LEMA1,I
0052 00030 003400      ERROR CCA
0053 00031 026026R      JMP ENDI
0054 00032 000001      TABLE DEC 1
0055 00033 177777      DEC -1
0056 00034 000001      DEC 1
0057 00035 000000      DEC 0
0058 00036 000000      DEC 0
0059 00037 000001      INDEX DEC 1
0060 00040 000001      CODE DEC 1
0061 00041 000000      BUF NOP
0062 00042 001457      NWORD DEC 815
0063          00043 000100          END
** NO ERRORS *TOTAL **RTE ASMB 92067-16011**

```

E.4 PROGRAM FOR ON-LINE PROCESSING

The purpose of the on-line processing is described in Chapter 2. The following is a listing of the program.

&SPMBM T=00004 IS ON CRO0013 USING 00056 BLKS R=0000

```

0001  FTN4, L
0002  $EMA(IBM, 5)
0003      PROGRAM SUPVR
0004      DIMENSION SF(20), NDXSF(20)
0005      DIMENSION IHEAD(15), IB1(4), RFQ(22)
0006  C    DIMENSION ITIME(10)
0007      DIMENSION POWR(184), NAM1(3), IFQ(2)
0008      DIMENSION IDCB(10)
0009      COMMON /IBM/ IBB(20480)
0010      COMPLEX CX
0011      COMMON LUMT, IHEAD, CX(400)
0012      EQUIVALENCE (IB1(1), CX(1)), (RFQ(1), IB1(4))
0013      EQUIVALENCE (IFQ(1), SF(1))
0014      EQUIVALENCE (AFRST, IDCB(3)), (ALAST, IDCB(5)), (DBMIN, IDCB(7)),
0015  *      (DBMAX, IDCB(9)), (NPTS, POWR(1)), (PMAX, POWR(2))
0016      DATA IDCB(1), DBMIN, DBMAX/0, -30.0, 0.0/
0017      DATA NAM1/2HBF, 2HPL, 2HT /
0018      DATA IBUS2, IBUS3/128, 192/
0019      DATA IEX, IST/2HEX, 2HST/
0020      DO 15 K=1, 15
0021  15    IHEAD(K)=0
0022      LUMT=10
0023  10    WRITE(1, 100)
0024  100    FORMAT('NUMBER OF SAMPLES(POWER OF 2)?')
0025      READ(1, *) NS
0026      I=1
0027  20    IF(I-NS) 25, 40, 30
0028  25    I=I+1
0029      GOTO 20
0030  30    WRITE(1, 200)
0031  200    FORMAT('NUMBER OF SAMPLES MUST BE A POWER OF 2')
0032  *      'PLEASE ENTER A NEW VALUE')
0033      GOTO 10
0034  40    RNS=NS
0035      WRITE(1, 300)
0036  300    FORMAT('NUMBER OF HYDROPHONES, SPACING(METRES)?')
0037      READ(1, *) NH, D
0038      IF(NH .LE. 0) GOTO 10
0039      RNH=NH
0040      BS=RNS*RNH
0041      IF(BS .LE. 40960.0) GOTO 45
0042      WRITE(1, 400)
0043  400    FORMAT('CAPACITY OF MAP MEMORY BUS 2 EXCEEDED')
0044  *      'REDUCE NUMBER OF SAMPLES AND/OR HYDROPHONES')
0045      GOTO 40
0046  45    WRITE(1, 450)
0047  450    FORMAT('NUMBER OF BEAMS, INITIAL ANGLE & FINAL ANGLE(DEGREES)?')
0048      READ(1, *) NBMS, AFRST, ALAST
0049  C
0050  C    COMPUTE SPACE REQUIRED FOR CO-VARIANCE MATRICES
0051  C    AND STEERING VECTORS FOR A SINGLE FREQUENCY
0052  C
0053      SP=2.0*RNH*(NH+NBMS)-2*NBMS
0054      IF(SP+BS .LE. 40960.0) GOTO 46
0055      WRITE(1, 470)
0056  470    FORMAT('NO SPACE FOR CO-VARIANCE MATRICES & STEERING VECTORS')
0057      CALL EXEC(6)
0058  46    WRITE(1, 460)
0059  460    FORMAT('SAMPLING FREQUENCY?')
0060      READ(1, *) SFG
0061      DF=SFG/NS
0062      FQMAX=DF*(NS/2-1)
0063  50    M=40960.0-BS

```

```

0064      NN=2*NH*NH
0065 C      NN=SIZE OF EACH CO-VARIANCE MATRIX
0066      L=20
0067 60      WRITE(1,600)
0068 600      FORMAT('HOW MANY FREQUENCIES DO YOU WISH TO SELECT?')
0069      READ(1,*) NF
0070      IF(NF .LE. 0) GOTO 40
0071      IF(NF .LE. L) GOTO 70
0072      WRITE(1,700)
0073 700      FORMAT('MAXIMUM NUMBER OF FRQUENCIES EXCEEDED'/
0074      * 'PLEASE ENTER A NEW VALUE')
0075      GOTO 60
0076 70      RNF=NF
0077      WRITE(1,800) DF,FGMAX
0078 800      FORMAT('FREQUENCY RESOLUTION =',F8.2,' HZ'/
0079      * 'MAXIMUM FREQUENCY IS ',F6.1,' HZ'/
0080      * 'INPUT SELECTED FREQUENCIES USING FREE FIELD FORMAT'/
0081      * 'USE SPACE OR COMMA TO SEPARATE DATA ITEMS'/
0082      * 'TERMINATE EACH LINE(EXCEPT THE LAST ONE) BY '//
0083      * 'FIRST FREQUENCY SPECIFIED SHOULD BE THE ONE'/
0084      * 'FOR IMMEDIATE PROCESSING AND DISPLAY')
0085      READ(1,*) (SF(I),I=1,NF)
0086      DO 72 I=1,NF
0087      IF(SF(I) .GT. FGMAX) GOTO 73
0088      K=SF(I)/DF+0.5
0089      SF(I)=K*DF
0090      NDXSF(I)=K+1
0091 72      CONTINUE
0092      GOTO 74
0093 73      WRITE(1,480)
0094 480      FORMAT('MAXIMUM FREQUENCY EXCEEDED'/
0095      * 'PLEASE SELECT A NEW SET')
0096      GOTO 70
0097 74      WRITE(1,500)
0098 500      FORMAT('SPECIFY PROCESSING TYPE''1=CBF''2=CBS''3=MLM''4=MLS')
0099      READ(1,*) ITYPE
0100      IF(ITYPE .LT. 1 .OR. ITYPE .GT. 4) GOTO 74
0101      WRITE(1,1000)
0102 1000      FORMAT('NUMBER OF DATA ACQUISITIONS?')
0103      READ(1,*) NA
0104 C
0105 C      PASS FIRST BUFFER TO PLOT PROGRAM
0106 C      THEN SCHEDULE PLOT PROGRAM
0107 C
0108      NPTS=NBMS
0109      ILEN=2*(NBMS+3)
0110      ICLS=0
0111      CALL EXEC(20,0,IDCB,10,0,0,ICLS)
0112      CALL EXEC(10,NAM1,ICLS,ITYPE,IFQ(1),IFQ(2),ILEN)
0113 C
0114      WRITE(1,75)
0115 75      FORMAT('RUN-ID(1-8 CHARS)?')
0116      READ(1,76) (IHEAD(K),K=10,13)
0117 76      FORMAT(4A2)
0118 D      WRITE(6,950)
0119 D950      FORMAT(1H,'FREQUENCY      INDEX')
0120 D      DO 75 I=1,NF
0121 D      WRITE(6,980) SF(I),NDXSF(I)
0122 D980      FORMAT(1H,F9.1,I9)
0123 D75      CONTINUE
0124      WRITE(1,1100)
0125 1100      FORMAT('TO TERMINATE THE PROGRAM HIT ANY KEY'/
0126      * 'THEN GIVE THE COMMAND BR,SUPVR')
0127 C
0128 C      THE DATA ACQUISITION PROGRAM IS EXPECTED TO PLACE INFORMATION
0129 C      INTO THE FIRST NS*NH WORDS(32 BITS EACH) OF MEMORY BUS 2
0130 C      SPACE FOR COVARIANCE MATRIX IS ALLOCATED AT HIGH END OF BUS 2
0131 C      BUT LAST WORD IS LEFT FREE TO AVOID BUFFER ADDRESS OUTSIDE LIMITS
0132 C
0133      BSCV=NN
0134      BACV=81918.0-2.0*BSCV
0135      CALL MPOP(6)

```



```

0136 C
0137 C SET UP THE FOLLOWING CONSTANTS IN THE SCALAR TABLE
0138 C 50 = RNH*RNH, USED IN MLM & MLS CALCULATIONS
0139 C 51 = 1.0/(RNH-1.0), USED IN CALCULATING AVERAGE POWER
0140 C 52-53 = CMPLX(1.0,0.0), USED BY CSDOT
0141 C 54 = 1.0/(NA*NS*NS), USED TO NORMALISE CO-VARIANCE MATRIX
0142 C 55 = 100.0, USED TO SCALE RESULTS BEFORE PRINTING
0143 C
0144 C CX(1)=CMPLX(RNH*RNH, 1.0/(RNH-1.0))
0145 C CX(2)=1.0
0146 C CX(3)=CMPLX(1.0/(NA*RNS*RNS), 100.0)
0147 C CALL MPWST(50, CX, 6, 1)
0148 C
0149 C CALCULATION OF STEERING VECTORS
0150 C
0151 C PI=3.14159265
0152 C AINC=(ALAST-AFRST)/(NBMS-1)
0153 C A=AFRST
0154 C
0155 C BSSV=2.0*(RNH-1.0)*NBMS
0156 C BASV=BACV-2.0*BSSV
0157 D WRITE(6,86) BASV
0158 DB6 FORMAT(' STEERING VECTORS START AT BUS2 ADDRESS: ',F6.1)
0159 C LROW=NBMS
0160 C ROWL=LROW
0161 C
0162 C CONFIGURE FIRST ROW AS A COMPLEX VECTOR
0163 C AND FILL WITH THE CONSTANT (1.0,0.0)
0164 C
0165 C CALL MPCLB(21, BASV, ROWL, 1, NH-1, 0)
0166 C DO 210 I=1, LROW
0167 C CX(I)=1.0
0168 210 CONTINUE
0169 C CALL MPWDB(21, CX, 1, 1)
0170 C
0171 C CONFIGURE SECOND ROW AND INITIALIZE TO
0172 C EXP(J*2*PI*F*D/C*COS(ANGLE))
0173 C
0174 C BA=BASV+4.0
0175 C CALL MPCLB(21, BA, ROWL, 1, NH-1, 0)
0176 C K=1
0177 C CONST=2.0*PI*D*SF(1)/1500.0
0178 C DO 220 J=1, NBMS
0179 C F=CONST*COS(PI*A/180.0)
0180 C CX(K)=CMPLX(COS(F), SIN(F))
0181 C K=K+1
0182 C A=A+AINC
0183 220 CONTINUE
0184 C CALL MPWDB(21, CX, 1, 1)
0185 C
0186 C SET UP REMAINING ROWS OF MATRIX
0187 C ROW(K)=ROW(2)*ROW(K-1)
0188 C
0189 C CALL MPEGB(22, 21)
0190 C LST=NH-1
0191 C DO 240 I=3, LST
0192 C BA=BA+4.0
0193 C CALL MPCLB(23, BA, ROWL, 1, NH-1, 0)
0194 C CALL CVMUL(23, 0, 21, 0, 22)
0195 C CALL MPEGB(22, 23)
0196 240 CONTINUE
0197 C
0198 C SET UP COSINE TABLE ON BUS 3 FOR USE BY FFT FUNCTION
0199 C
0200 C CTS=NS/2
0201 C CALL MPCLB(30, 0, 0, CTS, 0, 1, 0)
0202 C CALL VCOST(30, CTS)
0203 C
0204 C SET LST TO NO. OF HYDROPHONES-1
0205 C SET RNK=NO. OF 16-BIT HALFWORDS PER COMPLETE COVARIANCE MATRIX
0206 C SET SNK=NO. OF 16-BIT HALFWORDS PER EXTRACTED COVARIANCE MATRIX
0207 C INITIALIZE BAC TO START ADDRESS OF FIRST COVARIANCE MATRIX
0208 C INITIALIZE DAC TO START ADDRESS OF FIRST EXTRACTED MATRIX
0209 C INITIALIZE BAQ TO START ADDRESS OF FIRST Q-VECTOR(COLUMN)

```

```

0210 C
0211 LST=NH-1
0212 RNK=4. 0*RNH*RNH
0213 SNK=4. 0*(LST*LST)
0214 BAC=BACV
0215 DAC=RNS
0216 BAG=BACV+4. 0*RNH*(RNH-1. 0)
0217 C
0218 C COMPUTE ADDRESSES OF FOLLOWING VECTORS
0219 C AR=ADDRESS OF FIRST COLUMN OF COVARIANCE MATRIX
0220 C AQ=ADDRESS OF Q COLUMN OF COVARIANCE MATRIX
0221 C AC=ADDRESS OF FIRST COLUMN OF EXTRACTED MATRIX
0222 C
0223 AR=BAC
0224 AQ=BAG
0225 AC=DAC
0226 D WRITE(6, 330) BAC, BAG, DAC
0227 D330 FORMAT(1H , 3F10. 1)
0228 C
0229 INIT=1
0230 KK=1
0231 K=1
0232 C
0233 C CONFIGURE STEERING VECTORS AS A SINGLE COMPLEX BUFFER
0234 C
0235 CALL MPCLB(1BUS2+7, BASV, 0. 5*BSSV, 1, 1, 0)
0236 C
0237 C CONFIGURE EACH ROW OF COVARIANCE MATRIX ON BUS 3( R OR INV(R))
0238 C AS A COMPLEX VECTOR USING BUFFER NUMBERS FROM 32 ONWARDS
0239 C
0240 DO 426 J=1, LST
0241 CALL MPCLB(1BUS3+31+J, AC+4*(J-1), RNH-1. 0, 1, LST, 0)
0242 426 CONTINUE
0243 CALL BFPRM(LOBF, NRCH, XF, IER)
0244 IF(IER .NE. 0) STOP 1
0245 C
0246 C PREPARE TO WRITE FIRST BLOCK ON MAG. TAPE
0247 C
0248 777 IHEAD(1)=1
0249 IB1(1)=NH
0250 IB1(2)=NBMS
0251 IB1(3)=NF
0252 DO 78 J=1, NF
0253 RFQ(J)=SF(J)
0254 78 CONTINUE
0255 RFQ(NF+1)=D
0256 RFQ(NF+2)=1500. 0
0257 C
0258 C SWITCH M. T. LU# FROM 7 TO 10 OR VICE VERSA
0259 C
0260 LUMT=17-LUMT
0261 CALL EXEC(3, LUMT+400B)
0262 CALL EXEC(2, LUMT, IHEAD, 527)
0263 IHEAD(1)=2
0264 C
0265 C
0266 C CLEAR THE COVARIANCE MATRIX BEFORE THE
0267 C NEXT CYCLE OF ACCUMULATION AND PROCESSING
0268 C
0269 77 CALL MPCLB(20, BACV, BSCV, 0. 1, 0)
0270 CALL VCLR(20)
0271 C
0272 C TT=0. 0
0273 DO 80 I=1, NA
0274 C CALL EXEC(11, ITIME)
0275 C CALL MBCOV(7, INIT, XF, IEND)
0276 C IF(IEND .LT. 0) GOTO 99
0277 CALL MBC2(20, LOBF, NRCH)
0278 C CALL EXEC(11, ITIME(6))
0279 C IDIFF=ITIME(8)-ITIME(3)
0280 C IF(IDIFF .LT. 0) IDIFF=IDIFF+60
0281 C IDIFF=IDIFF*60+ITIME(7)-ITIME(2)
0282 C RDIFF=IDIFF+(ITIME(6)-ITIME(1))*0. 01
0283 C TT=TT+RDIFF

```

```

0284      CALL COVAR(NS,NH,K,NDXSF,NF,XF)
0285      K=0
0286      INIT=0
0287      80      CONTINUE
0288      C      WRITE(6,81) TT
0289      C81     FORMAT(' M.T. READ TIME = ',F8.2)
0290      C
0291      C      CALL EXEC(11,ITIME)
0292      C
0293      C      CONFIGURE COVARIANCE MATRIX AS A REAL VECTOR THEN
0294      C      NORMALIZE & CONJUGATE BY NEGATING THE ODD PART
0295      C
0296      CALL MPCLB(20,BACV,BSCV,0,1,0)
0297      CALL VSMA1(20,54,20,0)
0298      CALL MPODD(8,20)
0299      CALL VNEG(8)
0300      C
0301      C      CONFIGURE FIRST COLUMN OF COVARIANCE MATRIX AS A REAL VECTOR
0302      C      SO COLUMNS CAN BE MOVED ONE BY ONE TO BUS 3 USING "VMOV"
0303      C
0304      CALL MPCLB(IBUS2+8,BACV,2.0*RNH,0,1,0)
0305      CALL MPCLB(IBUS3+9,RNS,2.0*LST,0,1,0)
0306      C
0307      C      CREATE FUNCTION LIST TO EXTRACT A COLUMN OF A COVARIANCE MATRIX
0308      C
0309      IF(KK.EQ. 0) GOTO 85
0310      CALL MPBFL(1)
0311      CALL VMOV(9,8)
0312      CALL MPSBA(8,2*NH)
0313      CALL MPSBA(9,2*LST)
0314      CALL MPEFL(1)
0315      C
0316      85     CALL MPFOR(1,1,LST,1)
0317      C
0318      C      PRINT THE COVARIANCE MATRIX JUST TRANSFERRED
0319      C
0320      IF(ISSW(0).LT. 0) CALL CVMPR(31,LST,AC,SF(1))
0321      C
0322      C      CONFIGURE THE FOLLOWING BUFFERS
0323      C      21 = RESULT OF INV(R).Q(COMPLEX)
0324      C      22 = RESULT OF R.B(COMPLEX) OR INV(R).B(COMPLEX)
0325      C      23 = RESULTS MATRIX(REAL)
0326      C
0327      BA=10.0*NBMS
0328      STEP=4.0*(RNH-1.0)
0329      CALL MPCLB(21,BA,RNH-1.0,1,1,0)
0330      CALL MPCSO(22,58,NH-1,1)
0331      CALL MPCLM(23,2.0*NBMS,4,NBMS)
0332      C
0333      C      CONFIGURE LOGICAL BUFFER CONSISTING OF REAL PARTS
0334      C      OF THE DIAGONAL ELEMENTS(EXCEPT LAST ONE)
0335      C      OF THE COVARIANCE MATRIX
0336      C      THEN COMPUTE AVERAGE SUM IN SCALAR 132
0337      C      AND TAKE RECIPROCAL OF THIS VALUE
0338      C
0339      CALL MPCLB(20,BACV,RNH-1.0,0,2*(NH+1),0)
0340      CALL SSUM(132,20,51)
0341      CALL SDIV(129,1,132)
0342      C
0343      C      CONFIGURE FIRST STEERING VECTOR
0344      C
0345      CALL MPCLB(25,BASV,RNH-1.0,1,1,0)
0346      C
0347      D      WRITE(6,415)
0348      D415   FORMAT(1H1)
0349      CALL MPCOL(26,1,23)
0350      C
0351      C      CONFIGURE G-VECTOR(COLUMN)
0352      C
0353      CALL MPCLB(24,AG,RNH-1.0,1,1,0)
0354      C
0355      IF(ITYPE.LE. 2) GOTO 425
0356      C

```

```

0357      MAP=MPCLA(IBUS3+31,AC,LST,LST,1,0)
0358      CALL CMINV(31,124,2)
0359 C
0360 C      CMINV GIVES -INV(R) SO CONFIGURE THE
0361 C      AREA AS A REAL BUFFER & MULTIPLY BY -1
0362 C
0363      CALL MPCLB(31,AC,2.0*(LST*LST),0,1,0)
0364      CALL VSMA1(31,3,31,0)
0365 C
0366 C      COMPUTE INV(R).Q IN SCALAR TABLE
0367 C      AND SAVE RESULT IN BUFFER NO. 21
0368 C
0369      DO 420 J=1,LST
0370      CALL CSDOT(54+2*J,J+31,52,24)
0371 420    CONTINUE
0372      CALL MPEQB(27,21)
0373      CALL MPTSB(27,56,LST)
0374 C
0375 C      CONFIGURE ROW VECTOR Q* AND
0376 C      COMPUTE Q*.INV(R).Q IN SCALARS 56-57
0377 C
0378      BA=AR+STEP
0379      CALL MPCLB(27,BA,RNH-1.0,1,NH,0)
0380      CALL CSDOT(56,27,52,21)
0381 C
0382 C      EXTRACT POWER OF REFERENCE PHONE TO SCALAR 126
0383 C
0384 425    CONTINUE
0385      CALL MPEQB(27,24)
0386      CALL MPSBA(27,LST)
0387      CALL MPTBS(27,126,1)
0388 C
0389 C      COMPUTE Q-Q*.INV(R).Q IN SCALAR 127
0390 C
0391      CALL SSUB(127,126,56)
0392 C
0393      IF(KK .EQ. 0) GOTO 448
0394      KK=0
0395 C
0396 C      CREATE FUNCTION LIST TO PERFORM PROCESSING FOR ONE BEAM
0397 C
0398      CALL MPBFL(2)
0399 C
0400      IF(ITYPE .GE. 3) GOTO 435
0401 C
0402 C      CONFIGURE FIRST ROW OF COVARIANCE MATRIX
0403 C      COMPUTE R.B IN SCALAR TABLE
0404 C
0405      DO 430 J=1,LST
0406      CALL CSDOT(56+2*J,J+31,52,25)
0407 430    CONTINUE
0408 C
0409 C      TAKE CONJUGATE OF STEERING VECTOR
0410 C
0411      CALL CVCNJ(25)
0412 C
0413 C      COMPUTE B*.R.B IN SCALARS 120-121
0414 C      COMPUTE B*.Q IN SCALARS 56-57
0415 C      THEN (SCLR(56)**2+SCLR(57)**2)/SCLR(126)
0416 C
0417      CALL CSDOT(120,25,52,22)
0418      CALL CSDOT(56,25,52,24)
0419      CALL SMUL(56,56,56)
0420      CALL SMUL(57,57,57)
0421      CALL SADD(56,56,57)
0422      CALL SDIV(56,56,126)
0423      CALL SSUB(121,120,56)
0424      GOTO 445
0425 C
0426 435    CONTINUE
0427 C
0428 C      COMPUTE INV(R).B IN SCALAR TABLE

```

```

0429 C
0430 DO 440 J=1,LST
0431 CALL CSDOT(56+2*J, J+31, 52, 25)
0432 440 CONTINUE
0433 C
0434 C TAKE CONJUGATE OF STEERING VECTOR
0435 C
0436 CALL CVCNJ(25)
0437 C
0438 C COMPUTE B*.INV(R).B IN SCALARS 122-123
0439 C COMPUT B*.INV(R).Q IN SCALARS 56-57
0440 C THEN (SCLR(56)**2+SCLR(57)**2)/SCLR(127)
0441 C
0442 CALL CSDOT(122, 25, 52, 22)
0443 CALL CSDOT(56, 25, 52, 21)
0444 CALL SMUL(56, 56, 56)
0445 CALL SMUL(57, 57, 57)
0446 CALL SADD(56, 56, 57)
0447 CALL SDIV(56, 56, 127)
0448 CALL SADD(123, 122, 56)
0449 CALL SDIV(122, 50, 122)
0450 CALL SDIV(123, 50, 123)
0451 C
0452 445 CONTINUE
0453 C
0454 C OUR 4 RESULTS ARE NOW IN SCALARS 120-123
0455 C SO TRANSFER THEM TO NEXT COLUMN OF MATRIX
0456 C
0457 CALL MPTSB(26, 120, 4)
0458 C
0459 CALL MPSBA(25, LST)
0460 CALL MPEFL(2)
0461 C
0462 448 CALL MPIBC(0)
0463 CALL MPFOR(1, 1, NBMS, 2)
0464 CALL MPRBC(0)
0465 C
0466 C AFTER EXECUTION OF FUNCTION LIST 2 FOR EACH BEAM
0467 C ALL THE STEERING VECTORS HAVE BEEN CONJUGATED
0468 C SO RESTORE THEM READY FOR THE NEXT CYCLE
0469 C
0470 CALL CVCNJ(7)
0471 C
0472 C PROCESSING FOR SELECTED FREQUENCY COMPLETE
0473 C FOR SELECTED COLUMN OF RESULTS MATRIX
0474 C MULTIPLY RESULTS BY 1.0/(AVERAGE POWER)
0475 C FIND MAXIMUM VALUE AND COMPUTE 1.0/MAX
0476 C THEN MULTIPLY EACH ELEMENT BY THIS VALUE
0477 C CONVERT THESE VALUES TO DECIBELS.
0478 C TRANSFER TO HOST & PLOT
0479 C
0480 CALL MPROW(26, ITYPE, 23)
0481 CALL VSMA1(26, 129, 26, 0)
0482 CALL SMAX(131, 26, 127)
0483 CALL MPRST(131, PMAX, 2, 1)
0484 CALL SDIV(127, 1, 131)
0485 CALL VSMA1(26, 127, 26, 0)
0486 CALL VLOGH(26, 31, 26)
0487 CALL MPRDB(26, POWR(4), 1, 1)
0488 J=3+NBMS
0489 DO 449 I=4, J
0490 POWR(I)=RNORM(POWR(I))-5.0
0491 449 CONTINUE
0492 CALL EXEC(20, 0, POWR, ILEN, 0, 0, ICLS)
0493 C CALL EXEC(11, ITIME(6))
0494 C WRITE(6, 446) (ITIME(J), J=10, 1, -1)
0495 C446 FORMAT(' TIME: ', 10I6)
0496 C
0497 IF(IFBRK(0)) 90, 77
0498 90 WRITE(1, 452)
0499 452 FORMAT(' EX = EXIT PROGRAM'' ST = SWITCH TAPES')
0500 READ(1, 453) IANS

```



```

0501 453  FORMAT(A2)
0502      IF(IANS .EQ. IEX) GOTO 99
0503      IF(IANS .NE. IST) GOTO 90
0504 C
0505 C      WRITE EOF ON M.T. & REWIND
0506 C
0507      CALL EXEC(3,LUMT+100B)
0508      CALL EXEC(3,LUMT+400B)
0509      GOTO 777
0510 99    CALL MPCLS(0)
0511      CALL EXEC(3,LUMT+100B)
0512      NPTS=-1
0513      CALL EXEC(20,0,POWR,2,0,0,ICLS)
0514      CALL EXEC(6)
0515      END
0516      END*

```

&CVMBI T=00004 IS ON CRO0013 USING 00017 BLKS R=0000

```

0001 FTN4,L
0002      SUBROUTINE COVAR(NS,NH,K,ISF,NSF,XF)
0003      DIMENSION ISF(50)
0004      COMMON LUMT,IHEAD(15),HRB(800)
0005      DATA IBUS2/128/,IBUS3/192/
0006      IF(K .EQ. 0) GOTO 100
0007      RNS=NS
0008      RNH=NH
0009      BS=RNS*RNH
0010      CVSIZ=2.0*RNH*RNH
0011      ISIZ=2*NH*NSF
0012      BACV=81918.0-2.0*CVSIZ
0013      ACM=RNS+4.0*RNH*NSF
0014 C
0015 C      SET SCALE FACTOR(XF) IN SCALAR 128
0016 C      RAW DATA IS MULTIPLIED BY XF BEFORE FFT
0017 C
0018      CALL MPWST(128,XF,1,1)
0019 C
0020 C      CONFIGURE THE FOLLOWING PERMANENT BUFFERS
0021 C      1 = CONSTANT COMPLEX VECTOR(CURRENT VALUE OF SCALARS 56-57)
0022 C      2 = DATA AREA AS A SINGLE REAL BUFFER
0023 C      29 = DATA AREA AS A REAL MATRIX
0024 C      10 = SOURCE BUFFER(REAL) FOR FFT
0025 C      11 = COSINE TABLE FOR FFT
0026 C      12 = COMPLEX BUFFER FOR RESULT OF FFT
0027 C      30 = MATRIX OF SELECTED FREQUENCIES
0028 C      28 = COVARIANCE MATRIX FOR SELECTED FREQUENCY
0029 C      16 = FIRST COLUMN OF COVARIANCE INCREMENT MATRIX
0030 C      17 = COVARIANCE INCREMENT(CONFIGURED AS REAL VECTOR)
0031 C
0032      CALL MPDCV(1,56,NH,1)
0033      CALL MPCLB(IBUS2+2,0,0,BS,0,1,0)
0034      CALL MPCLM(29,0,0,NH,NS)
0035      CALL MPCLB(IBUS3+10,RNS,RNS,0,1,0)
0036      CALL MPCLB(IBUS3+11,0,0,0.5*RNS,0,1,0)
0037      CALL MPCLB(IBUS3+12,RNS,0.5*RNS,1,1,0)
0038      CALL MPCLM(30,RNS,2*NH,NSF)
0039      CALL MPCLB(28,BACV,CVSIZ,0,1,0)
0040      CALL MPCLB(IBUS3+16,ACM,RNH,1,1,0)
0041      CALL MPCLB(IBUS3+17,ACM,CVSIZ,0,1,0)
0042 C
0043 C      CREATE FUNCTION LIST 14 TO MOVE ONE ROW TO BUS 3,
0044 C      PERFORM FFT AND MOVE RESULT BACK TO ORIGINAL ROW
0045 C
0046      CALL MPBFL(14)
0047      CALL VMOV(10,3)
0048      CALL FFTR3(12,10,11)
0049      CALL VMOV(3,10)
0050      CALL MPNXR(3,29)
0051      CALL MPEFL(14)

```

```

0052 C
0053 C   CREATE FUNCTION LIST 15 TO MOVE
0054 C   2 COLUMNS CORRESPONDING TO ONE FREQUENCY
0055 C
0056   CALL MPBFL(15)
0057   CALL MPEVN(15,13)
0058   CALL MPODD(14,13)
0059   CALL VMOV(15,4)
0060   CALL VMOV(14,5)
0061   CALL MPNXC(13,30)
0062   CALL MPEFL(15)
0063 C
0064 C   CREATE FUNCTION LIST 13 TO SELECT & MOVE
0065 C   TWO COLUMNS OF FREQUENCY DATA
0066 C
0067   CALL MPBFL(13)
0068   DO 25 I=1,NSF
0069     J=2*ISF(I)-1
0070     CALL MPCOL(4,J,29)
0071     CALL MPCOL(5,J+1,29)
0072     CALL MPXFL(15)
0073 25  CONTINUE
0074     CALL MPEFL(13)
0075 C
0076 C   CREATE FUNCTION LIST 12 TO COMPUTE INCREMENT FOR
0077 C   ONE COLUMN OF A COVARIANCE MATRIX
0078 C
0079   CALL MPBFL(12)
0080   CALL MPTBS(19,56,1)
0081   CALL CCVML(15,0,1,0,18)
0082   CALL MPSBA(15,NH)
0083   CALL MPEFL(12)
0084 C
0085 C   CREATE FUNCTION LIST 11 TO CALCULATE THE
0086 C   INCREMENTS & UPDATE A COVARIANCE MATRIX
0087 C
0088   CALL MPBFL(11)
0089 C
0090 C   MULTIPLY SELECTED COLUMN BY COMPLEX CONJUGATE OF EACH ELEMENT
0091 C   AND SAVE THE VECTOR SO FORMED IN BUFFER 15
0092 C
0093   CALL MPEGB(15,16)
0094   CALL MPFOR(1,1,NH,12)
0095   CALL VAD(28,17)
0096 C
0097   CALL MPEFL(11)
0098 C
0099 C   CREATE FUNCTION LIST 9 TO PERFORM COMPLETE PROCESSING
0100 C
0101   CALL MPBFL(9)
0102 C
0103 C   RE-DEFINE THE FOLLOWING BUFFER
0104 C   18 = FIRST COLUMN OF FREQUENCY MATRIX
0105 C
0106   CALL MPCLB(IBUS3+18,RNS,RNH,1,1,0)
0107 C
0108 C   CONVERT DATA TO REAL AND SCALE
0109 C
0110   CALL VFLT(2,39,20,0)
0111   CALL VSMA1(2,128,2,0)
0112 C
0113 C   DEFINE FIRST ROW OF DATA
0114 C   & PERFORM FFT ON EACH ROW
0115 C
0116   CALL MPROW(3,1,29)
0117   CALL MPFOR(1,1,NH,14)
0118 C
0119 C   DEFINE FIRST COLUMN OF SELECTED FREQUENCIES
0120 C   AND EXTRACT THE SELECTED FREQUENCY DATA
0121 C

```

```
0122      CALL MPCOL(13,1,30)
0123      CALL MPXFL(13)
0124      C
0125      C      SELECT FIRST COVARIANCE MATRIX
0126      C      AND UPDATE EACH COVARIANCE MATRIX
0127      C
0128      CALL MPEGB(19,18)
0129      CALL MPXFL(11)
0130      CALL MPEFL(9)
0131      C
0132      C      REDEFINE DATA AREA AS AN INTEGER BUFFER
0133      C
0134      100      CALL MPCLB(IBUS2+20,0.0,BS,2,2,0)
0135              CALL MPXFL(9)
0136              CALL MPCLB(31,RNS,1.0*ISIZ,0,1,0)
0137              CALL MPRDB(31,HRB,1,1)
0138              DO 150 J=1,ISIZ
0139              HRB(J)=RNORM(HRB(J))
0140      150      CONTINUE
0141              CALL EXEC(2,LUMT,IHEAD,2*ISIZ+15)
0142              IHEAD(1)=IHEAD(1)+1
0143      C
0144      RETURN
0145      END
0146      END$
```

U212997

INITIAL DISTRIBUTION

	Copies		Copies
<u>MINISTRIES OF DEFENCE</u>		<u>SCNR FOR SACLANTCEN</u>	
MOD Belgium	2	SCNR Belgium	1
DND Canada	10	SCNR Canada	1
CHOD Denmark	8	SCNR Denmark	1
MOD France	8	SCNR Germany	1
MOD Germany	15	SCNR Greece	1
MOD Greece	11	SCNR Italy	1
MOD Italy	10	SCNR Netherlands	1
MOD Netherlands	12	SCNR Norway	1
CHOD Norway	10	SCNR Portugal	1
MOD Portugal	5	SCNR Turkey	1
MOD Turkey	5	SCNR U.K.	1
MOD U.K.	16	SCNR U.S.	2
SECDEF U.S.	65	SECGEN Rep. SCNR	1
		NAMILCOM Rep. SCNR	1
<u>NATO AUTHORITIES</u>		<u>NATIONAL LIAISON OFFICERS</u>	
Defence Planning Committee	3	NLO Canada	1
NAMILCOM	2	NLO Denmark	1
SACLANT	10	NLO Germany	1
SACLANTREPEUR	1	NLO Italy	1
CINCWESTLANT/COMOCEANLANT	1	NLO U.K.	1
COMIBERLANT	1	NLO U.S.	1
CINCEASTLANT	1		
COMSUBACLANT	1	<u>NLR TO SACLANT</u>	
COMMAIREASTLANT	1	NLR Belgium	1
SACEUR	2	NLR Canada	1
CINCNORTH	1	NLR Denmark	1
CINCSOUTH	1	NLR Germany	1
COMNAVSOUTH	1	NLR Greece	1
COMSTRIKFORSOUTH	1	NLR Italy	1
COMEDCENT	1	NLR Netherlands	1
COMMARAIMED	1	NLR Norway	1
CINCHAN	1	NLR Portugal	1
		NLR Turkey	1
		NLR UK	1
		NLR US	1
		Total initial distribution	240
		SACLANTCEN Library	10
		Stock	30
		Total number of copies	280

U212997